Modeling Flow Behavior in Naturally Fractured Reservoirs

A DISSERTATION
SUBMITTED TO THE
DEPARTMENT OF MINERAL RESOURCES AND PETROLEUM ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES OF THE
UNIVERSITY OF LEOBEN, AUSTRIA,
IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
"DOKTOR DER MONTANISTISCHEN WISSENSCHAFTEN"

written by
M.Sc. Mohammad Taghi AMIRY

February 2014

Advisor: O.Univ.Prof. Dipl.-Ing. Dr.mont. Dr.h.c. Zoltán E. HEINEMANN

I declare in lieu of oath that I did this work by myself using only literature cited at the end of this volume.

Dipl.-Ing. Mohammad Taghi AMIRY

Leoben, January 2014

Acknowledgments

Firstly, I would like to thank em.O.Univ.Prof. Dipl.-Ing. Dr.mont. Dr.h.c. Zoltán E. Heinemann for his continuous advise and guidance throughout this work and the valuable experience and knowledge I have gained.

I would like to express my very great appreciation to Dr. Georg Mittermeir of Heinemann OiL GmBH Leoben who was a great support throughout the study. I thank Univ.Prof. John C. Davis for his invaluable comments and because he interrupted his own work whenever I needed help. I thank Univ.-Prof. Dr.rer.nat. Clemens Brand for his time and I also thank everyone else who helped me in any stage during my study.

I would like to thank the CEO of Heinemann OiL GmbH, Dr. Gábor Heinemann for the financial support of this study, as well as the staff of HOL and my special thanks are extended to OMV company for their financial support.

I would like to thank Univ.-Prof. Dipl.-Ing. Dr.techn. Wilhelm Brandstätter from the Chair of Petroleum Production and Processing at the University of Leoben for his support during my study.

Dedication

I dedicate this work to my mother who has been always with me, and to my father as my role model and surely to my beloved wife, **Nafise** who always believed in me and encouraged me to go on with her endless love and support.

Abstract

The simulation of Naturally Fractured Reservoirs (NFRs) has always been a challenging topic from the time of their discovery. One of the main problems in their simulation is calculating the matrix-fracture transfer which governs the dynamic behavior of the reservoir. This work is a thorough investigation of the transfer term. It will be shown that the conventionally practiced method to calculate the potential difference for displacement of one fluid by another is unreliable; however, when the saturation distribution is homogeneous (e.g. for single-phase expansion drive or solution gas dive), the conventional transfer equation can be reliably used. In 2004, Heinemann[44] suggested that recovery curves could be used to calculate the matrix-fracture transfer term. This research focuses on this idea contributing to its theoretical foundation and practical implementation.

A workflow for using recovery curves for simulation of NFRs is (1) field investigations, (2) generating the recovery curves for the matrix blocks, (3) lumping (weighted-averaging of) matrix block recovery curves to generate the simulation cell recovery curves, (4) using the recovery curves in the mathematical model and finally, (5) utilizing recovery curves in the simulation model.

New concepts are devised and different tools and methods that were necessary are designed and developed which include:

- “**Single Matrix Block Analysis**” method and software tools are developed that allow detailed study of matrix blocks with different fluid and rock properties under various initial and boundary conditions. Each investigation produces a recovery curve and average saturation versus time plots.
- “**Matrix Block Classes**” are defined that allow scaling the recovery curve of matrix blocks with different porosities, permeabilities, and even different shapes by use of a theoretically-derived “**dimensionless time**” scaling factor.
- “**Recovery Curve Regions**” are defined on the full field. For each such region and distinct drive mechanism a “**lumped recovery curves**” represents the overall recovery from all the different matrix blocks in the simulation cell.
- “**Using Recovery Curves instead of the Matrix-Fracture Transfer Function**”.

Several challenges existed for each of these steps. They were successfully overcome and the developed methodology was implemented in an industrial reservoir simulator. The new concept can be easily implemented in any mature dual-porosity reservoir simulator by making only minor extensions to it. Thus they can also benefit from the additional accuracy of the recovery curve method without reducing the industrial practicability of the simulators.

Kurzfassung

Bereits mit der Auffindung einer natürlich geklüfteten Kohlenwasserstofflagerstätte stellt deren Simulation eine Herausforderung dar. Eines der Hauptprobleme ist die Berechnung des Matrix-Kluft Transferterms, welcher das dynamische Verhalten der Lagerstätte bestimmt. Diese Studie ist eine sorgfältige Untersuchung des Transferterms. Die Unzulänglichkeit der Potentialdifferenzberechnung - praktiziert bei der konventionellen Methode - für die Verdrängung eines Fluids durch ein anderes wird gezeigt; ist jedoch eine homogene Sättigungsverteilung (wie z.B. bei der Einphasenexpansion bzw. dem Gasentlösungstrieb) gegeben, ist die konventionelle Transfergleichung zuverlässig. Im Jahr 2004 regte Heinemann[44] an, Entölungskurven für die Berechnung des Matrix-Kluft Transferterms zu verwenden. Diese Forschungsarbeit konzentriert sich auf diese Idee und trägt zu ihrem theoretischem Fundament und ihrer praktischen Anwendbarkeit bei.

Ein Arbeitsablauf für die Simulation von solchen Lagerstätten gliedert sich in (1) Felduntersuchungen, (2) Generierung der Entölungskurven für die Matrixblöcke, (3) Zusammenfassung (gewichtete-Mittelung) von Matrixblockentölungskurven zu Simulationszellenentölungskurven, (4) Verwendung dieser Kurven im mathematischen Modell und schlußendlich (5) Anwendung der Entölungskurven im Simulationsmodell.

Neue Konzepte wurden erdacht und unterschiedliche Werkzeuge und die dafür notwendigen Methoden wurden entworfen und entwickelt. Diese beinhalten:

- Es wurde die "**Einzelmatrixblockanalyse**" Methode und Softwareanwendung entwickelt, welche eine detailierte Untersuchung des Matrixblocks unter Berücksichtigung verschiedenster Fluid- und Gesteinseigenschaften sowie Randbedingungen ermöglicht. Eine Entölungskurve und die mittlerer Sättigung als eine Funktion der Zeit sind das Ergebnis dieser Untersuchungen.
- "**Matrixblockklassen**" werden eingeführt, welche die Skalierung der Entölungskurven eines Matrixblockes für verschiedenen Porositäten, Permeabilitäten und sogar verschiedener Matrixblockformen mithilfe eines theoretisch abgeleiteten "dimensonslosen Zeitfaktors" erlauben.
- "**Entölungskurvenregionen**" werden für das Gesamtmodell definiert. Für jede dieser Regionen und eine bestimmte Triebart repräsentieren "**Zusammengefasste Entölungskurven**" die Gesamtentölung aller Matrixblöcke einer Simulationszelle.
- **"Verwendung von Entölungskurven statt der Matrix-Kluft Transferfunktion."**

Für jeden der hierin erklärten Schritte bestanden zahlreiche Herausforderungen. Diese wurden erfolgreich überwunden und die entwickelte Methodology wurde in einem industriellen Lagerstättensimulator umgesetzt. Das neue Konzept kann einfach in jedem gereiften Doppelporositätslagerstättensimulator zum Einsatz gebracht werden. Lediglich wenige Erweiterungen sind notwendig. Somit können auch diese von der zusätzlichen Genauigkeit der Entölungskurvenmethode profitieren, ohne Reduzierung der industriellen Praktikabilität des Simulators.

Table of Contents

Modeling Flow Behavior in Naturally Fractured Reservoirs

Chapter 1

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

Chapter 7

Chapter 8

Appendix A

Appendix B

Appendix C

Appendix D

Appendix E

List of Figures

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

Appendix A

Appendix B

Appendix C

Appendix D

Appendix E

List of Tables

Chapter 5

Chapter 6

Chapter 1

Introduction

1.1 Motivation of the Work

1.1.1 Background

Naturally Fractured Reservoirs (NFRs) play an important role in petroleum industry and the global economy. Natural *fractures* exist practically in all reservoirs, dividing the reservoir rock in pieces, called "*matrix blocks*". In this case one must distinguish between matrix and fracture porosities and permeabilities. The reservoir is a single porosity one when it does not need special considerations and the reservoir rock can be regarded as a single continuum if the fractures are isolated or the matrix is tight. The reservoir cannot be modeled as a single continuum if the matrix has reservoir quality and the fractures are interconnected, building more or less extended networks. However, accurately simulating the naturally fractured reservoirs is still a real challenge as it has ever been. For fractured reservoirs, obtaining the correct data and forecasting the reservoir performance are much more difficult than for conventional reservoirs. In order to design an appropriate development plan, one needs a credible reservoir description that includes fracture maps in terms of size, orientation, connectivity, conductivity, and frequency distribution. This information has to be turned into a reliable fracture network characterization.

The multiple-continua concept is generally used to model NFRs, where typically two continua, the matrix and the fracture, overlap. The main problem is then to find the right description of the fluid exchange between them while modeling a recovery process. Mathematically, this leads to the extension of the flow equation by the so-called *matrix-fracture transfer term*. It can consider some or all of the acting forces such as the force of compression, chemical potential (diffusion), gravitational force, capillary forces and viscous forces with or without hysteresis or alteration of the properties (e.g. wettability). By considering all factors, the mathematical description of the matrix-fracture transfer becomes so complex that the corresponding mathematical model can be only numerically handled. An ideal approach would be to use a Discrete Fracture Network (DFN) giving the frame and the boundary conditions for the discretized matrix blocks individually. Such an approach tries to model the conductive fractures and matrix blocks based on statistically generated fracture distribution maps. Physical properties such as transmissivity or storage, and geometrical properties such as size, elongation and orientation are assigned to each (matrix or fracture) element based upon measured data or geologically conditioned statistical distributions derived from the measured values.

However, gathering these additional data is quite difficult and more importantly, the required calculation cost (in terms of CPU-time and memory) for large carbonate fields (such as the ones existing in the middle east) makes this approach industrially senseless. Academic toys can be created but it is difficult to imagine how they could be coupled with the geologic reality. Heinemann believes[42] that the application of such an approach in connection with the current industrial praxis and future needs is neither possible nor necessary.

In another approach, it is possible to upscale the discrete fracture model to a continuous one, and to reduce the matrix conglomerate to a cluster of identical blocks. This is then the Warren and Root[99] sugar cube concept. A fracture cell is now connected to an overlapping matrix cell and the mass transfer is calculated based on the average pressures, saturation and composition at both sides. The transfer between the matrix and the fracture is calculated using a simplified scheme, considering two points, i.e. two control volumes.

To summarize, modeling NFRs appropriately is much more complicated than conventional reservoirs, especially for industrial use. The most commonly used methods in petroleum industry are dual-porosity and/or dual-permeability approaches in which the reservoir is modeled as two overlapping domains: the matrix domain and the fracture domain.

1.1.2 Suggested Improvements

One of the major problems in simulation of NFRs conventionally, is calculation of the matrix-fracture fluid transfer term. Heinemann[44] suggested that it is possible and advantageous to use recovery curves to calculate the matrix-fracture transfer term. Pirker[81] proved its applicability on a column model with only one recovery curve. The goal of this work is to apply the recovery curve method in a commercial industrial simulator overcoming all the challenges existing in that direction. The deficiencies of the other existing dual-porosity approaches, in short, are the following:

For the conventional dual-porosity/dual-permeability approach (used in industry):

- Unrealistic and nonphysical modeling of matrix blocks representing several matrix blocks as a single simulation cell.
- Over-homogenization of the matrix blocks to levels that makes the results inacceptable.
- Inadequacy of defining a common transfer function that can accurately predict how all the matrix blocks in a simulation cell exchange fluid with the fracture.
- Oversimplifying physical phenomena that occur in the model.

and for the DFN approach:

- Very difficult to measure the parameters necessary to calculate the grid and relying on statistics.
- Currently impossible to be used for large-scale reservoirs and too far away from being industrially applicable to full fields.

This work is an attempt to use the recovery curve method in the industry for simulation of naturally fractured reservoirs, with the following criteria:

- should assure a considerably higher physical reality than the currently applied conventional approach,
- should capture the heterogeneity of the matrix domain,
- should be practical and *readily* usable in industry with a clearly specified workflow,
- should be, unlike DFN, not too costly in terms of computational time and should be industrially applicable with today's computational technology.

"The reason of the emphasis on achieving a *ready-to-use* method is that all university campuses are paved with aborted ideas and wasted efforts of former PhD graduates. To avoid this fate, the author searched for solutions only, which are usable in connection with industrial standard procedures and tools. Looking back, none of the earlier engineering achievements are still viable or part of mankind's scientific heritage. Engineering Sciences seemingly do not create permanent values. No work should be invested in something which is not timely."[1] Engineering research must concentrate on the actual problems and must lead to "ready-to-use" results:

- The new solution should not require supplementary research or development before applying it.
- The new solution should fit today's existing technology.
- The new solution should be developed on real cases and not only demonstrated on simple models (such as the SPE comparative examples). Academic examples always work; real field cases seldom do! The extension of a solution from academic problems to the real world is never straight forward as researchers often like to declare so.

1.2 Objectives

The objectives of this work are therefore the following:

- achieving a higher physical reality than the conventional dual-porosity model,
- while considering the high heterogeneity of the matrix domain, by taking into account the distribution of different matrix properties (e.g. porosity, permeability and shape factor) even within a single simulation cell,
- not as CPU-costly as DFN, for the increase of the accuracy; some extra preliminary effort is needed to create the recovery curves, but the tools to help perform this task with as little effort as possible are also developed and applicable,
- and more importantly, propose a method that is practical at the time-being and can be implemented in all currently used industrial reservoir simulators with minimum efforts.

1.3 The Approach

In this work, it is proposed to simulate naturally fractured reservoirs using the matrix recovery curve to calculate the matrix-fracture transfer term (instead of using the transfer function) in the conventional dual-porosity/permeability model. This method is validated theoretically and also implemented in a commercial reservoir simulator and tested on small scale test models and a real reservoir.

The recovery curves can be obtained from any source. The procedure to measure it in the laboratory for a piece of matrix rock is described (as well as the detailed method to measure its shape factor). A software tool is also designed and implemented that can model the laboratory experiment and output the recovery curve for a matrix block from its shape factor (and the boundary conditions).

The proposed dimensionless time conversion factor is used to scale the recovery curves and to classify the matrix block IDs into matrix block classes. The distribution of matrix block classes in the reservoir, helps defining the recovery curve regions.

In the simulation of naturally fractured reservoirs using recovery curves, one recovery curve region is assigned to a simulation cell (like any other property such as PVT region number) and the recovery curve(s) associated with this number is used to calculate the matrix-fracture transfer rate. The details of this method as well as the definitions of all the mentioned terms here, are explained in Chapter 3 and Chapter 4.

The applicability of the proposed methods are always regarded as the topmost priority (along with its theoretical validity). In other words, if a proposed academic method is too accurate to be ever applicable in the industry, it will remain just as an *academic* and *research* method like hundreds of such investigations that are proposed yearly around the world. However, the value of a research (especially in the petroleum industry) is much more emphasized when it can actually be applied in the industry. Therefore, the suggested methods in this work, while improving the accuracy of the simulations way better than the conventional dual-porosity model, will not make it impractical or impossible for industrial application.

1.4 Outline

Chapter 1 **Introduction**

This is the current chapter which contains a brief introduction to the objectives and contents of this dissertation.

Chapter 2 **State of the Art of Modeling Naturally Fractured Reservoirs**

In this chapter a brief literature-review of the relevant methods of simulating naturally fractured reservoirs is presented.

Chapter 3 **The Elements of the New Approach**

In this chapter some concepts, ideas and theories are presented as separate headlines. All of the mentioned concepts in this chapter are related to the new presented approach of simulating naturally fractured reservoirs using recovery curves, that will be explained in details in Chapter 4. The elements covered in ~~the~~ this chapter, are all developed/improved/suggested in this work to overcome the problems of creating and using recovery curves for dual-porosity simulation listed as follows:

- Shape Factor,
- Calculation of the Shape Factor,
- Single Matrix Block Analysis,
- Investigation of the Transfer Term,
- Matrix Block Identification Methods,
- Calculation of the Matrix-Fracture Transfer from a Recovery Curve,
- Lumping Recovery Curves.

Chapter 4 **Theory of the Work**

In this chapter, the "*new concept for modeling and simulating naturally fractured reservoirs using recovery curves*" suggested by this research is presented and explained in details. The following terms will be here only defined, but their practical application will be in Chapter 5 presented:

- Simulation Cell Model,
- Dimensionless Time,
- Matrix Block IDs,
- Matrix Block Classes,
- Recovery Curve Regions.

Chapter 5 **Recovery Curve Approach to Simulation of NFRs**

In this chapter, the workflow for the new approach of using recovery curves for calculating the matrix-fracture transfer to be used in the dual-porosity concept and the preparatory steps are explained in detail:

- Identifying Matrix Blocks and Their Distribution,
- Generating Recovery Curves for Matrix Block IDs,
- Classifying Recovery Curves,
- Identifying Recovery Curve Regions,
- Using Region Recovery Curves for Simulation of NFRs,
- Advantages and Limitations.

Chapter 6 Verification and Case Study

This chapter verifies the proposed method on a case study using a column model. Amongst other tests, this case study also includes the simulation results of the newly proposed recovery curve method and comparing them to the conventional approach.

Chapter 7 Conclusions

In this chapter, the findings and conclusions from this work are briefly presented.

References

This chapter lists all the references for this work (including the Appendices).

Appendix A The Mathematical Model

In this chapter, mathematical modeling of naturally fractured reservoirs is briefly explained.

- Dual-Continua Model,
- Matrix-Fracture Transfer Term.

Appendix B Derivation of a Generalized Shape Factor

This chapter describes the derivation of Heinemann and Mittermeir[48] generalized shape factor which is used in this work as the basis for calculations and recommended to be used industrially as well.

Appendix C Laboratory Experiment of Measuring Recovery Curve

The recovery curve of any piece of rock can be measured in the laboratory with a method similar to what is explained in this chapter. Following scenarios are considered:

- Water Imbibition
- Gas Gravity Drainage

Furthermore, other Experiments and Considerations are discussed.

Appendix D Implementation of Single Matrix Block Analysis

Single Matrix Block (SMB) analysis is an attempt to numerically simulate the laboratory experiment of studying the matrix block depletion under different conditions and recovery mechanisms.

The laboratory experiment is described in Appendix C and the theory about the proposed SMB model is explained in Section 3.4. In this chapter, the details about the implementation of the proposed Single Matrix Block Analysis model in a commercial reservoir simulator, ***PRS***[45], is explained.

Appendix E Production Mechanisms in Naturally Fractured Reservoirs

In this chapter, the mechanisms that take part in extraction of hydrocarbons from the matrix to the fracture are described:

- Expansion and Solution Gas Drive,
- Capillary Imbibition,
- Gravity Drainage,
- Viscous Displacement,
- Molecular Diffusion.

1.5 Scientific Achievements and Technical Contributions

1.5.1 New Scientific Ideas Presented

- The shape factor is a physical parameter and should not be treated as a meaningless tuning parameter for history-matching. (Section 3.2.4)
- The boundary conditions for the matrix-fracture transfer calculation are fluid saturations and not phase potentials. (Section 3.4.8)
- Defining "Recovery Curve Regions" for calculation of the matrix-fracture transfer term in different parts of the reservoir. (Section 4.3.4, Section 5.2.1.4)

1.5.2 Scientific Achievements

- It is demonstrated that the conventional transfer equation is valid for single-phase and solution gas expansion drives and is questionable under displacement processes. (Section 3.5)
- It is demonstrated that the recovery curve can be used to calculate the matrix-fracture transfer in the conventional dual-porosity model. (Section 3.8)
- A theoretical definition of a dimensionless time is derived which successfully allows use of a dimensionless recovery curve to represent the recovery of a wide range of matrix block IDs (of different shape factor, porosity, permeability, oil compressibility and viscosity values) as long as the drive mechanism is the same. (Section 3.10)
- A method is devised to consider the effect of viscous pressure gradient in calculation of the single matrix block recovery. (Section 3.4.8.4, Section 3.4.8.5)
- A detailed method to measure and calculate the shape factor for a matrix block is presented. (Section 3.3)
- Considering the phase segregation assumption in calculating the gravitational effect is unrealistic and is one source of complication in calculation in the conventional dual-porosity model; (Section 3.5.3) the proposed recovery curve approach considers the gravitational effect without phase-segregation assumption. (Section 3.4.8.3.1)

1.5.3 Technical Contributions

- Developing Single Matrix Block Analysis software tool to calculate the recovery curves for the matrix blocks of any ID from anywhere in the reservoir model. (Section D.3.2)
- Developing a data repository software for handling and storing all the recovery curves for different matrix block IDs in a reservoir and performs lumping them into region curves. (Section D.4.2)
- Integrating the Single Matrix Block Analysis tool in a commercial reservoir simulator to allow this analysis on actual reservoirs easily just by a single line of input in the full field model. (Section D.3)
- Implementing the use of recovery curves for calculation of the matrix-fracture transfer term (instead of the conventional transfer function) in a commercial reservoir simulator. This allows using the recovery curves for calculation of the current reservoir models without any hassle.
- A standard model for calculation of the single matrix block recovery curve is presented. (Section 3.4)
- The single matrix block analysis has to be performed in parallel to the full field simulation to have control and insight over the physical processes. (Section 3.4)

1.5.4 Publications of the Author

1. Amiry, M.T.: “Importance and Applicability of a Generalized Shape Factor by Modeling Dual Porosity Reservoirs”, Conference paper, ECMOR XIII, Sep. 2012.
2. Amiry, M.T., Heinemann, Z.E.: “The Heinemann-Mittermeir Generalized Shape Factor and Its Practical Relevance”, accepted to print in Journal of Chemical and Petroleum Engineering, 2014.
3. Amiry, M.T.: “Heinemann-Mittermeir Shape Factor: Measurement and Significance of Anisotropy in Modeling Dual-Porosity Reservoirs”, Hungarian Journal of Mining and Metallurgy OIL and GAS, March 2014.
4. Amiry, M.T, Heinemann, Z.E.: “Limitations of the Conventional Transfer Function and the Practical Implications”, prepared for Erdöl Erdgas Kohle journal, 2014.

Chapter 2

State of the Art of Modeling Naturally Fractured Reservoirs

2.1 Introduction

In this chapter the state of the art of modeling naturally fractured reservoirs is presented briefly. Some parts that are be used in this work are explained in more details.

2.2 Characterization

During the characterization process of naturally fractured reservoirs a sound understanding of the fractures, the fracture network and its influence on the fluid flow behavior should be gained. Based on this knowledge realistic geomodels are constructed that provide the full field parameters needed for dynamic simulation. The modeling itself, faces some inherent problems as the range of scales on which the data are measured is so broad, the fracture types vary and the fracture network is complex. A state of the art fracture modeling workflow consists of the following steps:

1. Definition of the fracture sets.
2. Fracture indicator identification and derivation.
3. Identification and ranking of fracture drivers.
4. Neural network fracture modeling.
5. Creation of discrete fracture network realization.
6. Fracture property modeling.

Probably, the main challenge in construction of reliable fracture models is generally the limited amount of data and the difference in scale on which the data are measured, e.g. as small as cores taken from the well up to 3D seismic attributes calculated for the entire field. Besides, it is difficult to obtain representative samples of the reservoir because of the extreme heterogeneity of fractured rocks. Some questions arising during the characterization process of a naturally fractured reservoir might be:[68]

- Which of the many fractures in a well affect the fluid flow and are important?
- How is the diverse data from different sources integrated in the fracture network characterization (e.g. core analysis of one well and well logs in another)?
- Which strategy should be used to interpolate the fracture information through a given volume (e.g. the reservoir volume)?
- How to translate and simplify the fracture information for a reservoir simulator?

Consequently, not single deterministic models are constructed, but several equally probable realizations will be constructed which need to be screened and verified by dynamic simulation runs.

In order to design an appropriate development plan (dynamic reservoir model), one needs a credible reservoir description that includes all fracture related properties such as:

- size of the fractures,
- number of fractures per unit length (fracture intensity),
- fracture orientation and spatial organization,
- fracture aperture,
- fracture porosity,
- fracture conductivity,
- fracture connectivity, and
- fracture relative permeabilities.

The parameters mentioned above are inferred from direct (e.g. core analysis) and indirect sources (e.g. well logging methods) of measurement to get the required information for the fracture characterization process.

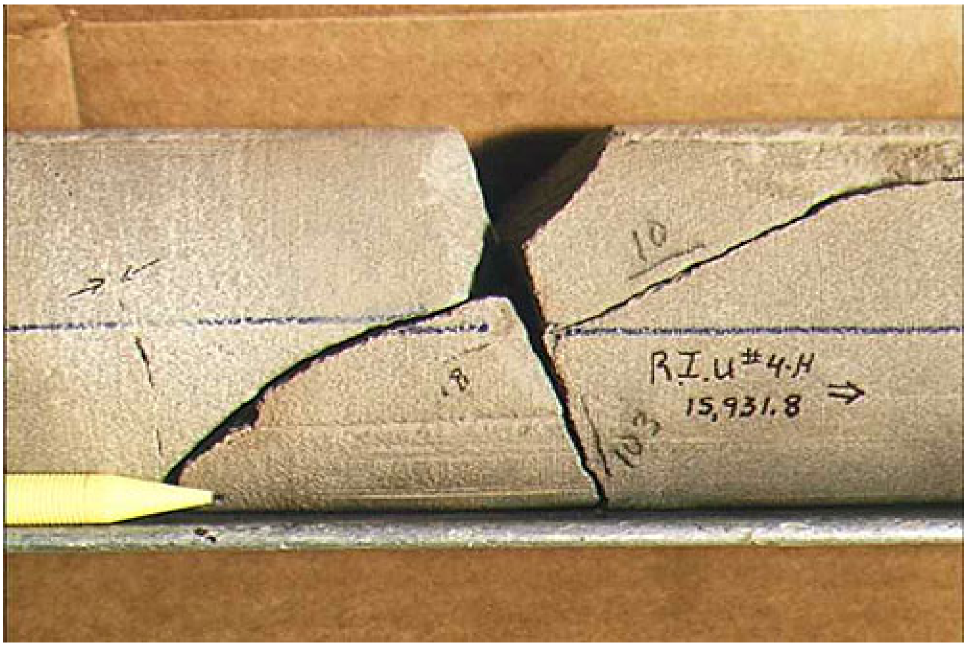


Figure 2.1: Fractured Core[102]

2.2.1 Characterization methods

Various options of fracture detection are presented in the literature[15],[32],[68],[77],[79]. The question is how to gather all the information into a complete reservoir description. When facing the problem of modeling a fractured reservoir, oil and gas companies have three available approaches:

The first one is the *geomechanical approach* where it is attempted to reconstruct the tectonic history of the fractured reservoir. Unfortunately, all the existing tools based on this approach use over-simplistic models in which the complex geology of the reservoir is ignored and homogenous and isotropic rock properties are assumed in the calculations. Furthermore, the end result of this approach is a strain map which mostly is very similar to a simple curvature map, easily derived from the current structural surfaces. In addition to the inability of this approach to account for the complex and heterogeneous geology of all fractured reservoirs, there is no room to incorporate any 3D seismic attribute in the geomechanical process.

The second approach commonly used to model fractured reservoirs is the *Discrete Fracture Network (DFN*) where the reservoir volume is filled with fractures that are represented by planes or disks. For many years, the DFN models lacked geological realism since they were randomly distributed in the reservoir, ignoring the fact that fracture density at any point is affected by the thickness of the reservoir at that point, its lithology and porosity, its proximity to faults and numerous other geologic drivers.[79] Since the introduction of DFN models, there was a need for constraining the realizations to some geologic input. Attempts have been made to control the fracture generation with some indicators. However, these attempts used one single geologic driver and ignored the others, and most importantly did not account for the complex interplay of the drivers. Ouenes and Hartley[73] introduced the concept of *conditioned DFN* whereas a large effort is spent to integrate all the geological drivers in a continuous fracture model which is then used to constrain the DFN models. A field example illustrating this approach is given by Zellou[103].

Finally, the third approach uses a *continuous framework* where many geological drivers could be incorporated in creating an integrated fracture model. These continuous fracture models stem from the simple observation that fracture intensity depends on many geological drivers (the most commonly known being the structural setting, proximity to a fault, lithology, porosity and thickness). Because all these drivers and their complex interaction must be accounted for during the modeling process, Ouenes and Hartley[73] use a regular 3D grid model similar to the one used in geological modeling or in seismic cubes along with a collection of artificial intelligence tools to create truly integrated fractured reservoirs. The approach described in detail by Ouenes and Feng[72] was successfully used in various fields and basins and one of its most striking advantage is its ability to integrate seismic data in the modeling process.

2.2.2 Shape Factor Distribution

The fracture intensity is the number of fractures per unit length of a sample line. Therefore, it estimates the distance between the individual fractures. The shape factor is directly related to the fracture intensity, since it is determined by the size of the matrix blocks in the dual-porosity system, which depends on the fracture intensity.[81]

During the characterization process of the naturally fractured reservoirs, a statistical distribution of the fracture intensity is obtained. Figure 2.2 shows an example of such distribution. Zones of high fracture intensity are in red, those with low fracture intensity are in purple. In order to assess the recovery for a given field a single average value for the fracture intensity can be calculated. In order to assess the correct range of the expected recovery and its probability realistically not only a single average value but rather the calculated distribution should be utilized. Because of the fact that there is a direct relationship between fracture density and shape factor, the distribution of the fracture intensity can be transformed to the corresponding distribution of the shape factor.[81]

Different statistical distributions of the fracture parameters can be found in the literature. For example, a negative exponential distribution for the fracture length was published by Panek[74], by Miller[63] or by Bloomfield[12]. Rouleau and Gale[90] assume exponential distribution; Gudmundsson[40] and Jamison[51] have found power law distributions of different fracture characteristics (length, spacing). Some authors[12],[25],[39],[63],[74],[87] found the fracture spacing, and some authors[6],[12],[25],[34],[39],[63], the fracture length to be lognormal distributed.

Ultimately, these parameters can be used to estimate the shape factor distribution in the reservoir which demonstrates how matrix and fracture interact in different parts of the reservoir (for example using FracaFlow[71] software tool).

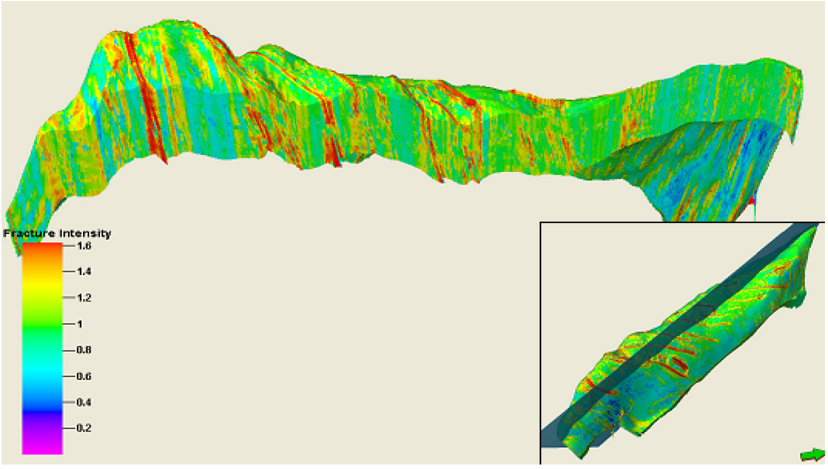


Figure 2.2: Fracture intensity distribution

2.3 Discrete Fracture Modeling

The idea behind Discrete Fracture Modeling (DFM) is that the fracture planes are represented individually and the fluid flow behavior can be predicted from the fracture geometries and data on the transmissivity of individual fractures.[77] This methodology combines "a forward approach based on geosciences and an inverse approach based on reservoir engineering"[14] utilizing the mapping of fracture planes in 3D space to develop an interconnected fracture network. To perform a field-scale simulation, average reservoir properties such as conductivity, anisotropy and storage capacity are required and therefore, discrete fracture information such as size, orientation, location and spacing are used in building DFMs. Unlike the previous approaches, DFM depends upon a very accurate fracture network description, without an emphasis on any given type of information, and is best suited for those situations where the flow behavior is dominated by significant fractures. Information about fractures and geological setting is not only used to construct the models, but also to constrain the models, ensuring that a realistic geological model is produced. The final outputs of DFM are the equivalent fracture parameters which can be used in any fractured reservoir simulator. These parameters are[14] effective permeabilities, matrix block size, fracture porosity.

2.3.1 Discrete Fracture Modeling Methods

Different methods exist to create discrete fracture models (DFM). Some authors[56],[69] use fracture geometry data (location, orientation and size) with boundary element or finite element methods to derive the so-called "*equivalent permeability tensor of the discrete fracture network*". Geostatistics is another tool which has been used to estimate the large-scale hydraulic conductivity, also based on fracture geometry data.[17] Alternative DFM methods[7],[13],[27],[92] have the advantage of being implemented as stochastic models, in which multiple realizations provide a quantitative measure for uncertainty and variability.[26] Figure 2.3 demonstrates the fracture network where each plane represents one single fracture and the colors represent the fracture set that it belongs to.

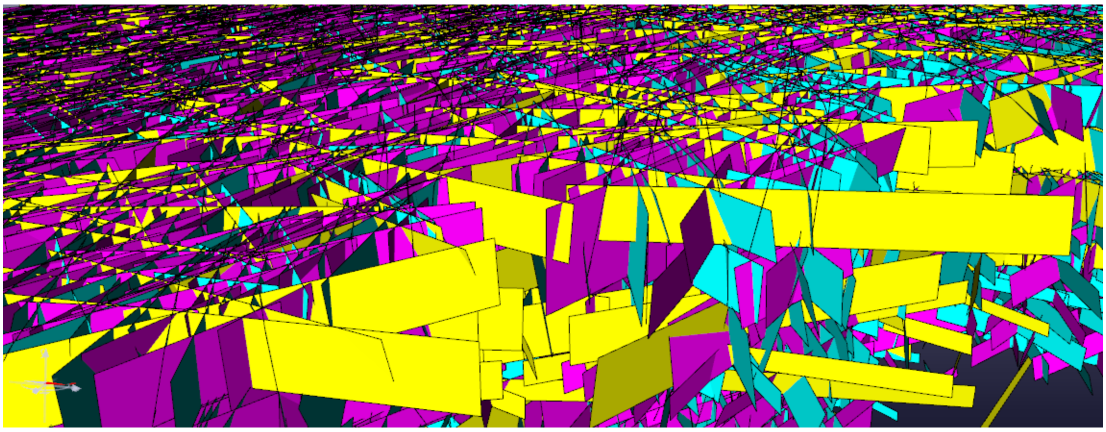


Figure 2.3: Representation of a DFN (by courtesy of Franlab[71])

2.4 Conventional Modeling of Naturally Fractured Reservoirs

The foundation of the dual-porosity model was laid down by Barenblatt et al.[8] and Warren and Root[99] fifty years ago. The following era has seen many modifications to the basic dual-porosity model by various researchers. In this approach, NFRs are thought to comprise of an interconnected fracture system that provides the main flow paths (with high permeability and low storage volume) and the reservoir rock (or matrix) that acts as the main source of hydrocarbons (which has low permeability and high storage volume).[10] Thus, it is the matrix system that contains most of the petroleum, but the production to the wells is through the high-permeability fracture system, implying that it is the matrix-fracture interaction that mainly controls the fluid flow. The NFR is modelled with two overlapping domains: the *matrix domain* and the *fracture domain*. A sample model of such reservoir (in Cartesian grid) is illustrated in Figure 2.4(a). There are two identically shaped domains. The reservoir consists of grid cells or *simulation cells* in each domain.

This means that at the position of the same grid cell, there exists a *matrix cell* (in the matrix domain) and a *fracture cell* (in the fracture domain). Each cell can have neighboring cells from (up to) six directions in its own domain plus one (or more, in some models) extra neighbor in the other domain that is the overlapping cell from that domain.

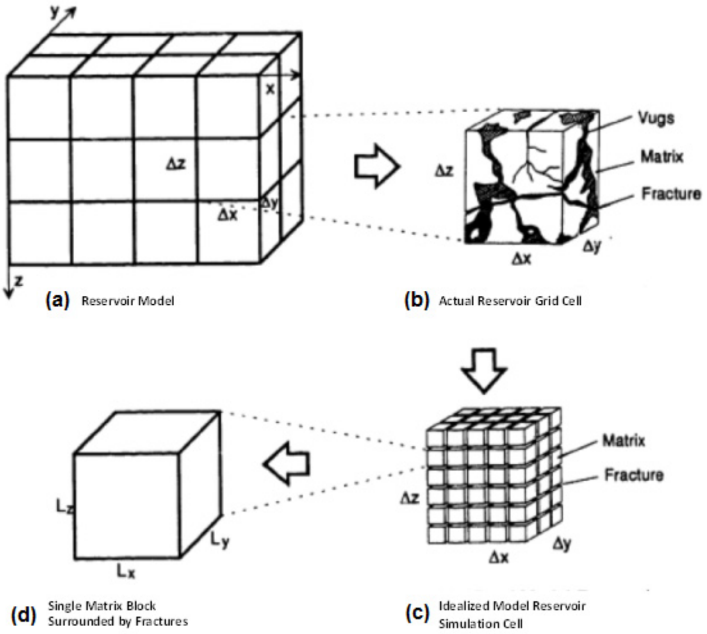

y
x
z
Δz
Δx
Δy
(a) Reservoir Model
Vugs
Matrix
Fracture
Δz
Δx
Δy
(b) Actual Reservoir Grid Cell
Matrix
Fracture
Δz
Δx
Δy
(c) Idealized Model Reservoir Simulation Cell
Lz
Ly
Lx
(d) Single Matrix Block Surrounded by Fractures

Figure 2.4: Warren and Root[99] Model

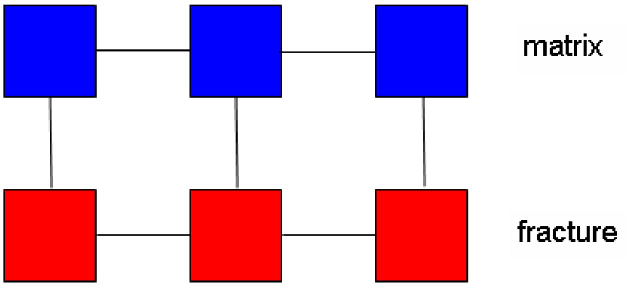


Figure 2.5: Dual-porosity dual-permeability concept

In other words, it is assumed that two continua exist at the same location, both fill the space completely without gaps. Each porous material has its own variables. The matrix properties and variables are signed by the subscript *m* and the fracture by *f*. For example the porosities are ϕ_m and ϕ_f which are functions of the pressures p_m and p_f, respectively. The pore space is occupied by fluids and separated by phase surfaces. The proportion of a phase, *p*, in the pore volumes are defined as the phase saturations S_{pm} and S_{pf}. Figure 2.5 shows such a model (in which the matrix and fracture domains are actually overlapping, but are displayed separated to show the neighborhood relations). The combination of the matrix cell and fracture cell displayed in Figure 2.5, is equivalent of the Figure 2.4(c) cell. It is necessary to clarify some ambiguities in nomenclature and distinguish between matrix *blocks* and simulation *cells*:

Figure 2.6 is a two-dimensional outcrop of a naturally fractured formation. In order to model it numerically, the space has to be discretized. The red squares in Figure 2.6 (or cubes in 3D) represent the control volumes, in other words the "*simulation cells*".

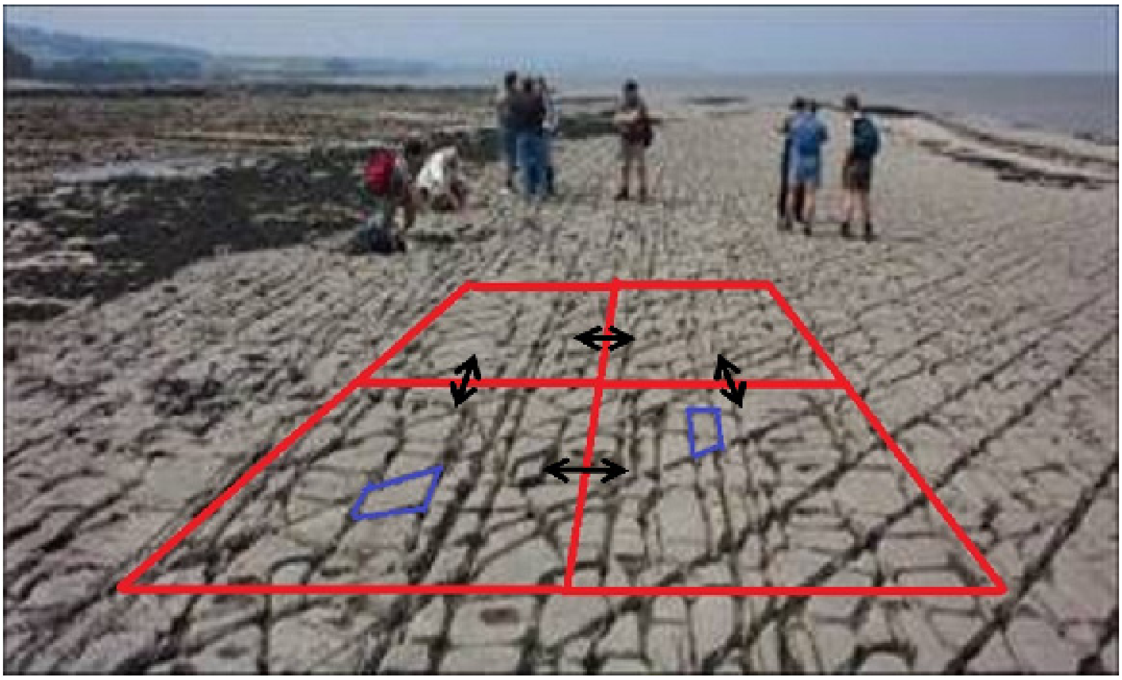


Figure 2.6: Simulation cell vs. Matrix block, (by courtesy of Roxar training[91])

The dual-continua approach considers two simulation cells at each location (i.e. at the place of each red square), one is the "*matrix cell*" and the other is the "*fracture cell*". This divides the space into two identically shaped overlapping domains: the "*matrix domain*" and the "*fracture domain*". The fracture domain, consisting of fracture cells, usually has a much higher permeability and less porosity than the matrix domain. This means that the fracture cells can exchange fluids easily with their neighboring fracture cells (along the black double-arrows in Figure 2.6). Matrix cells, may or may not be modelled to exchange fluids with their matrix neighbors (along the same black double-arrows) resulting in dual-permeability or dual-porosity models respectively.

2.4.1 Single Porosity Model Mathematical Formulation

The molar continuity equation in a *single continuum space* can be written for component *c* from Equation 2.1. The first term on the left-hand side in Equation 2.1 is the *convection term* or *flow term*. It gives the moles of component *c* flowing through the unit surface per unit time (molar velocity). The summation over all phases is necessary, because the particular component may be present in several phases. The second term is the *diffusion term*. It gives the moles of the component *c* moving through the unit of surface per unit time, driven by the concentration gradient. The third term is the *volumetric source/sink term*. It determines how much of component *c* will be taken out, or is supplied to the unit volume per unit time. The right-hand side is the *accumulation term*. The summation over all phases and the integral over the whole control volume gives the accumulation of component *c* within this control volume.

$$-\nabla\left(\sum_{p=1}^{P}\lambda_{pc}D_p x_{pc}\bar{\bar{k}}\nabla\Phi_p\right)-\nabla\left(\sum_{p=1}^{P}\overrightarrow{J_{pc}}\right)+q_c=\frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P}S_p D_p x_{pc}\right)\quad,\qquad(2.1)$$

where,
$\underline{P}$ is the number of phases,
$\bar{\bar{k}}$ is the permeability tensor [m^2],
λ_p is the mass mobility of component *c* in phase *p* [kmol/m^3/cp],
D_p is the specific molar density [kmol/m^3],
x_{pc} is the mole fraction of component *c* in phase *p* [-],
J_{pc} is the molar flux of the component *c* in phase *p* [kmol/s/m^2],
q_c is the molar source/sink of the component *c* [kmol/s/m^2],
S_p is the saturation of phase *p* [-].

2.4.1.1 Permeability Tensor

"The permeability tensor is symmetric and positive definite per definition. Some upscaling methods can produce a non-symmetrical permeability tensor, which would mean that the Darcy flow has shear effects. For such an effect, no physical evidence exists."[49]

The asymmetric permeability tensor is solely the product of the upscaling method. The permeability tensor from Equation 2.1 can be split into a *normalized dimensionless* tensorial part and an effective scalar *apparent permeability* k_a. Therefore, the permeability tensor of Equation 2.1 can be rewritten as:

$$\bar{\bar{k}} = \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{12} & k_{22} & k_{23} \\ k_{13} & k_{23} & k_{33} \end{pmatrix} = k_a \bar{\bar{k}}' = k_a \begin{pmatrix} k'_{11} & k'_{12} & k'_{13} \\ k'_{12} & k'_{22} & k'_{23} \\ k'_{13} & k'_{23} & k'_{33} \end{pmatrix} . \qquad (2.2)$$

k_a can be any number to calculate the new tensor values as $k'_{ij} = k_{ij}/k_a$. This step is to separate the apparent permeability, which is more locally variant and the anisotropy tensor, which tends to be valid for larger scales.

However, when calculating the actual matrix-fracture interflow, this value will be multiplied again in the tensor to result in the actual k_{ij} directional permeability values (in Darcy) as will be shown in Equation 2.8.

Mathematically speaking, this can be regarded as a simple factorization of the value k_a (along with its dimension) from the tensor to normalize the tensor and to make it dimensionless. It has been tried by different researchers[18],[23],[60] over the years to give physical definitions for k_a and k'_{ij} tensor values; what is common, is that they all suggest that for the matrix apparent permeability k_a the following inequality should hold:

$$\kappa_b \le k_a \le \kappa_a \ , \qquad (2.3)$$

where κ_b is the harmonic mean and κ_a is the arithmetic mean of k_{ij}*'s* in Equation 2.1. But as mentioned, *theoretically*, this number can be any value greater than zero. It is suggested to use the *maximum principle permeability value*, $k_{x,}$ (Equation 2.4) or the *equivalent isotropic permeability* after Muskat[67] (Equation 2.5):

$$k_a = k_x \ , \qquad (2.4)$$

$$k_a = \sqrt[3]{k_x k_y k_z} \ , \qquad (2.5)$$

where k_x, k_y, k_z are the permeability values in *x*-, *y*- and *z*-principle permeability directions respectively. The *principle permeability direction* is explained in Section 3.3.8.

2.4.2 Dual-Porosity Balance Equation

In the dual-porosity model (described in more details in Appendix A), it is assumed that a single-porosity flow occurs in the fracture domain; and in dual-permeability model also an independent single-porosity flow occurs in the matrix domain. Consequently, Equation 2.1 should be duplicated, once with the subscript *m* and once with the subscript *f*.

The fluid transfer between the two otherwise-independent domains is considered through a molar flow term called, the *matrix-fracture transfer term*, q_{cmf} which is the amount of fluid transfer, in moles, for each fracture cell to/from the same cell in the matrix domain. Therefore, the balance equation (Equation 2.1) will be written as Equation 2.6 for the fracture and Equation 2.7 for the matrix.

$$\nabla\left(\sum_{p=1}^{P}\Lambda_{pc}k_a\bar{\bar{k}}'\nabla\Phi_p\right)_f+\nabla\left(\sum_{p=1}^{P}\phi S_p D_p \mathrm{D}_{pc}\nabla x_{pc}\right)_f+q_{cf}-q_{cmf}=\frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P}S_p D_p x_{pc}\right)_f \tag{2.6}$$

$$\nabla\left(\sum_{p=1}^{P}\Lambda_{pc}k_a\bar{\bar{k}}'\nabla\Phi_p\right)_m+\nabla\left(\sum_{p=1}^{P}\phi S_p D_p \mathrm{D}_{pc}\nabla x_{pc}\right)_m+q_{cm}+q_{cmf}=\frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P}S_p D_p x_{pc}\right)_m \tag{2.7}$$

2.4.3 Production Mechanisms in NFRs

In NFRs, it is mainly the matrix-fracture fluid interaction which governs the production from the reservoir, therefore, the production mechanisms are focused on how the matrix is depleted and the petroleum is transferred from the matrix to the fracture (where it will be transported to the producing wells). The importance of each production mechanism differs from a single-porosity system, since the interaction of the two systems (matrix and fractures) has to be taken into account. The dominant forces in the two distinct systems can also be quite different. The relative importance of the different production mechanism is basically determined by the shape of the matrix blocks and by rock and fluid properties of the system. In the conventional dual-porosity model the production mechanisms in naturally fractured reservoirs are modelled to calculate a so-called "transfer term" which represents the matrix-fracture fluid transfer rate. These main production mechanisms, described in details in Appendix E, are:

- Fluid expansion and solution gas drive
- Imbibition
- Gravity drainage
- Viscous displacement
- Molecular diffusion

2.4.4 Matrix-Fracture Transfer

Kazemi et al.[52] extended Warren and Root[99] formulation for multi-phase flow and numerically solved the dual-porosity system in three dimensions and derived the transfer term:

$$q_{cmf} = V_{cell}\sigma k_a \sum_{p=1}^{P} \Lambda_{pc}\bar{\bar{k}}'(\Phi_{pf} - \Phi_{pm}), \tag{2.8}$$

where

V_{cell} is cell pore volume [m^3],

k_a is the apparent permeability of the matrix block [D],

σ is the shape factor [m^{-2}],

Λ_{pc} is the mass mobility of component c in phase p [kmol],

$\bar{\bar{k}}'$ is the dimensionless normalized anisotropy tensor,

Φ_{pf} is the fracture potential [bar],

Φ_{pm} is the matrix potential [bar].

Shape factor was first introduced by Barenblatt et al.[8] (see Section 3.2.3) who did not discuss its physical meaning other than to point out that it was inversely proportional to the square of some "*characteristic length*" of the matrix block. Since then, numerous equations for the shape factor have been proposed for various block shapes. The most commonly assumed shape has probably been for a cube of length L. For cubical blocks, Warren and Root[99], based on an obscure derivation suggested that $\sigma = 60/L^2$. Kazemi et al.[53] suggested $\sigma = 12/L^2$ based on a finite-difference approximation to the flow equations with the entire matrix block represented by a single finite-difference cell. Coats[21] suggested the value $\sigma = 24/L^2$. Quintard and Whittaker[85] used volume averaging to conclude that $\sigma = 49.62/L^2$. Zimmerman et al.[104] used Fourier analysis with transient flow assumption and derived the result $\sigma = 3\pi^2/L^2$. This value was confirmed by Lim and Aziz[58], and was re-derived by Mathias and Zimmerman[61] in the Laplace domain. Kazemi-Gilman-Elsharkawy[52] suggested a generalized pseudo-steady state shape factor, valid for all possible irregular matrix block shapes:

$$\sigma = \frac{1}{V_m}\sum_{j=1}^{N}\frac{A_j}{\left|\vec{d}_j\right|}, \tag{2.9}$$

where V_m is the volume of the matrix block, A_j is the area of a surface open to flow and d_j is the distance from the open surface to the centroid of the matrix block. Note that in this shape factor, it is assumed that the permeability (k_a) is isotropic and the same in all directions.

Heinemann and Mittermeir[48] used a control volume finite-difference discretization on an anisotropic dual-continuum of irregular shape (presented in Appendix B) and generalized Kazemi et al.[52] shape factor and mathematically proved that it is exact under pseudo-steady state conditions for anisotropic matrix blocks as well:

$$\sigma = \frac{1}{V_m} \sum_{j=1}^{N} A_{ij} \frac{\left|\overline{\overline{k'}}\hat{n}_j\right|}{\left|\vec{d}_j\right|}, \tag{2.10}$$

where $\overline{\overline{k'}}$ is the permeability tensor and $\hat{n}_j$ is the unit normal vector of each surface open to flow.

The wide range of values that have been proposed for the shape factor can, in part, be explained by the fact that as have been known for many decades (cf. de Swaan[28]), Equation 2.8 is only accurate in the pseudo-steady state regime. Since, Equation 2.8 is not valid over all times and conditions, there is no unambiguous way to define the most appropriate value of *σ*. De Swaan[29] proposed that *σ* be chosen so as to render Equation 2.8 accurate at the time at which the (mean) pressure change in the matrix block has attained 50% of its eventual value, for the case of a step-change in the fracture pressure. This approach has some practical advantages, but causes the model to no longer be asymptotically accurate at large times.

In this work, the transfer function is put to test (see Section 3.5) and it will be shown that unlike the general assumption over the decades, it is not the shape factor which is the cause of the discrepancy in the calculation of the matrix-fracture transfer term, but actually ignoring the effects of saturation distribution when calculating the driving forces in the conventional model.

The deficiencies of the conventional transfer equation has been known for a long time and many authors have tried to tackle this problem by different approaches, none of which have been applicable enough to be vastly practiced in industrial scale. Here, two of such approaches are briefly explained. In the first approach, the matrix cells are gridded finely in an attempt to consider the effect of inhomogeneous saturation distribution in the transfer, but it will be shown that it fails to consider all the recovery mechanisms appropriately while it increases the calculation time several times. The second approach, is an attempt to redefine the transfer function to consider much more details, but the process is so complex that it is virtually impossible to imagine it would ever get the chance to be industrially applicable either.

2.4.4.1 The MINC method

In 1985, Pruess and Narasimhan[83] proposed the Multiple INteracting Continua (MINC) method for numerical simulation of heat and multiphase fluid flow in multidimensional fractured media. The MINC method is able to handle the transient flow of heat and fluids between matrix and fractures numerically. This is in contrast to the standard dual-porosity model which uses an analytical expression, under the assumption of pseudo-steady state flow for coupling the flow between matrix and fracture. Thus, the MINC model can be regarded as a generalization of the Barrenblatt *et al.*[8] and the Warren and Root[99] models.

For most cases the pseudo-steady state flow description is sufficient.[94] Only for some cases, such as a well test in a dual-porosity system, the transient nature of the matrix-fracture flow is of importance for reservoir simulation.

The traditional dual-porosity approach assumes that the matrix can be accounted for as a single simulation cell in the matrix domain. But in the MINC approach it is different, as it assumes two kinds of grid:

1. A primary grid, which is the discretization of the reservoir volume into simulation cells.
2. A secondary grid, which is formed by nested matrix sub-cells that represent each matrix cell (in the primary grid).

Such a MINC grid is illustrated in Figure 2.7. Figuratively speaking, each matrix grid cell could be regarded as a "Russian Doll" of nested sub-cells. Even though the MINC method is a generalization of the dual-porosity model, it is based on some assumptions:

1. The fractured reservoir is represented by uniform cubes, representing the matrix, which are bounded by fracture planes on all faces. A generalization to arbitrary fracture geometries is available, using a proximity function.
2. The depletion process is influenced by two aspects: First, the global movement of fluid and heat through the reservoir, mainly taking place in the fracture system. Second, fluid and heat flow between the matrix and the fracture. Thermodynamic conditions, such as temperatures, pressures, vapor saturations, in the matrix depend primarily on the distance from the nearest fracture. This leads to a discretization of the matrix into sub-volumes, where the interfaces between the matrix volume elements are parallel to the nearest fracture.
3. The equipotential surfaces of pressure, temperature and other thermodynamic state variables are assumed to be planes parallel to the nearest fracture. This assumption will hardly be the case for real flow problems. Nonetheless, some conditions can be imagined where a "distance-only" approximation leads to an accurate prediction of interporosity flow:
   - 3.1 Initial thermodynamic conditions in the matrix depend approximately only on the distance from the fracture. This includes the special case of uniform initial conditions in the matrix blocks.
   - 3.2 The boundary conditions for the matrix blocks have to stay approximately constant at all times. If the transient conditions taking place in the fractures occur in a certain way this condition will be satisfied.

However, the MINC method can be correctly applied to more than the latter conditions. As soon as the pressure and respectively the temperature gradient, calculated with the distance-only approximation leads to correct total flow rates over a sub-cell interface area, the MINC method can be regarded as accurate. With the distance-only approximation of MINC pressure, viscous and capillary forces can be handled properly, but gravity force, because of its directional dependency, would be problematic.

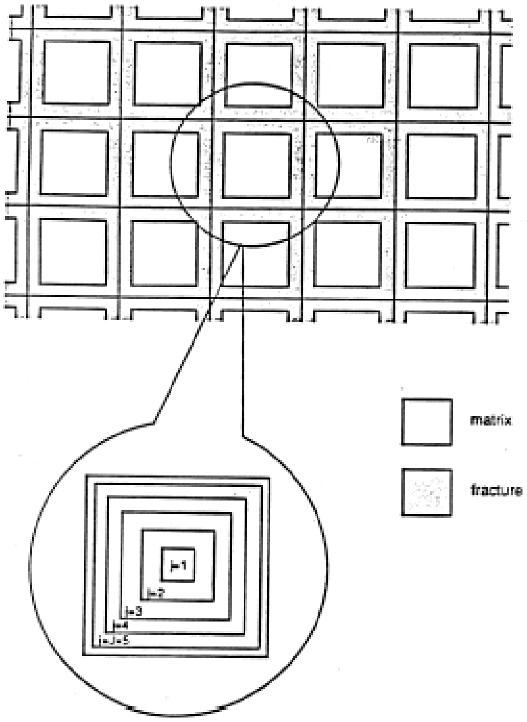


Figure 2.7: The MINC descritization

The grid generation for the MINC method can be described in three steps:

1. In the first step, the primary mesh has to be defined. This may be done by considering global reservoir geometry, global variations of reservoir properties and the desired spatial grid resolution. The global mesh divides the reservoir into N elemental volumes V_n, where the subscript n ranges from 1 to N.
2. In the second step, the geometric parameters of the nested volume elements, V_j, within the primary volumes, V_n, have to be determined. In order to do this, the volume fraction, f_j, of each individual sub-cell has to be known (specified) upfront. The variable j is valid from 1 to J, where 1 refers to the fracture and J corresponds to the innermost matrix sub-cell.
   The volume fraction f_1 can be easily derived from the fracture spacing, L, and the fracture aperture, δ. Thus leading to:

$$f_1 = \frac{[L^3 - (L - \delta)^3]}{L^3} \approx \frac{3\delta}{L}. \tag{2.11}$$

   The other volume fractions can be chosen arbitrarily, only the constraint of $\sum_{j=1}^{J} f_j = 1$ has to be fulfilled.
   The region closest to the fractures is of special interest.

As a consequence, the outer matrix cub-cells having the indices $j = 2,3,...$ will be more closely spaced than the elements farther from the fracture with the indices $j = J, J\text{-}1,...$. This can be also seen in Figure 2.8.

Based on the fracture spacing, L, and the volume fraction of the j^{th} continuum, f_j, the sub-cell volume can be calculated by

$$V_j = f_j L^3 \ . \tag{2.12}$$

Details on how the nodal distances and the interface areas are defined, as well as the used discretization scheme, integral finite difference formulation, for the MINC method can be found in the publication by Pruess and Narasimhan[83].

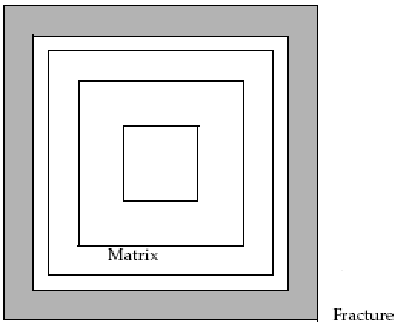


Figure 2.8: Nested matrix sub-cells with different spacing[95]

3. Usually the fracture spacing, L, is much smaller than the size of the grid cells of the primary grid. Thus Pruess and Narasimhan[83] formulated scaling laws, how to partition the volume elements, V_n, of the primary grid into a sequence of interacting continua elements, V_{nj}, where j is indexing a continuum within V_n.

In a simulation grid cell with nested volume elements as outlined above, only one flow direction will be possible. This will be an outward directed flow, from the matrix to the fracture, as production of reservoir fluids causes a pressure decline in the fracture system. This imposes a limitation to the MINC method. For instance, compositional problems arising in EOR operations require a modification of the method which would allow flow through the matrix.

ECLIPSE[95] offers a method similar to the MINC method. Three different geometry options, with increasing dimensional complexity, are available, a linear (1D), a cylindrical (2D) and a spherical (3D) one. The cylindrical and the spherical geometries are approximated by square cylinders and cubes, respectively.

2.4.4.2 Defining very complex transfer functions

Some authors have tried to redefine the transfer functions considering more complexity to achieve a higher level of accuracy than the conventional transfer function. However, these attempts nearly always lead to transfer functions that are so complex that makes them virtually impractical and useless anywhere else but the academic case for which they were originally developed. In this section, the final form of the suggested transfer function by Sarma[93] is briefly mentioned. Sarma[93] suggests rewriting the material balance equation, on each face of the (cubic) matrix cell separately, as shown in Equation 2.13 and Equation 2.14 respectively for the matrix and fracture domains:

$$\left\{\left\{\sum_{s=1}^{ns}\left[T_s\sum_p \lambda_p\rho_p X_{cp}\Delta\Phi_p\Big|_s\right] + WI^W \cdot \sum_p \lambda_p\rho_p X_{cp}(p-p^W)\right\}^{n+1}\right\}_m - \tau_{c_{mf}} = \left\{V\frac{\phi^{n+1}\sum_p S_p\rho_p X_{cp}\Big|^{n+1} - \phi^n\sum_p S_p\rho_p X_{cp}\Big|^n}{\Delta t}\right\}_m \tag{2.13}$$

$$\left\{\left\{\sum_{s=1}^{ns}\left[T_s\sum_p \lambda_p\rho_p X_{cp}\Delta\Phi_p\Big|_s\right] + WI^W \cdot \sum_p \lambda_p\rho_p X_{cp}(p-p^W)\right\}^{n+1}\right\}_f - \tau_{c_{mf}} = \left\{V\frac{\phi^{n+1}\sum_p S_p\rho_p X_{cp}\Big|^{n+1} - \phi^n\sum_p S_p\rho_p X_{cp}\Big|^n}{\Delta t}\right\}_f \tag{2.14}$$

where,

p is the subscript for different phases,
c is the subscript for each component,
s is the subscript for each cell face,
W is the superscript for the well,
n is the superscript for the current time step,
$n+1$ is the superscript for the next time step,
λ_p is the mobility of phase p [1/cp],
ρ_p is the phase density of phase p [lbm/ft^3],
X_{pc} is the mole fraction of component c in phase p [-],
ϕ is the porosity [-],
Φ_p is the potential of phase p [psia],
V is the cell volume [ft^3],
S_p is the saturation of phase p [-],
WI is the well index [mD.ft],
p is the pressure [psia],
T is the transmissibility [md.ft],

$\tau_{c_{mf}}$ is the matrix-fracture transfer term for component *c*, calculated from Equation 2.15:

$$\tau_{c_{mf}} = Vk_a\sigma_{PD}\sum_p[\lambda_p\rho_pX_{cp}(\Phi_{pm}-\Phi_{pf})] - V\phi_m\sigma_{SD}\sum_p[\rho_pX_{cp}(S_p-S_{pi})] \tag{2.15}$$

where,

$_i$ is the subscript for initial property value,
k_a is the matrix apparent permeability [mD],
σ_{PD} is the shape factor for pressure diffusion [1/ft^2],
σ_{SD} is the shape factor for saturation diffusion [1/day].

Equation 2.15 consists of two different terms: the pressure diffusion term and the saturation diffusion term which are meant to take different recovery mechanisms into consideration. For each one of them, a different time-dependent shape factor function is used which have to be separately calculated from fine-grid simulation of the matrix block under the specified conditions before actually being able to use the values in the conventional dual-porosity simulation.

Figure 2.9 shows the fine-grid model to calculate the best fitting function values for the “pressure diffusion shape factor” from Equation 2.16 and the “saturation diffusion shape factor” from Equation 2.17.

$$\sigma_{PD} = f(t)\,, \tag{2.16}$$

$$\sigma_{SD} = bt^{-m}\,, \tag{2.17}$$

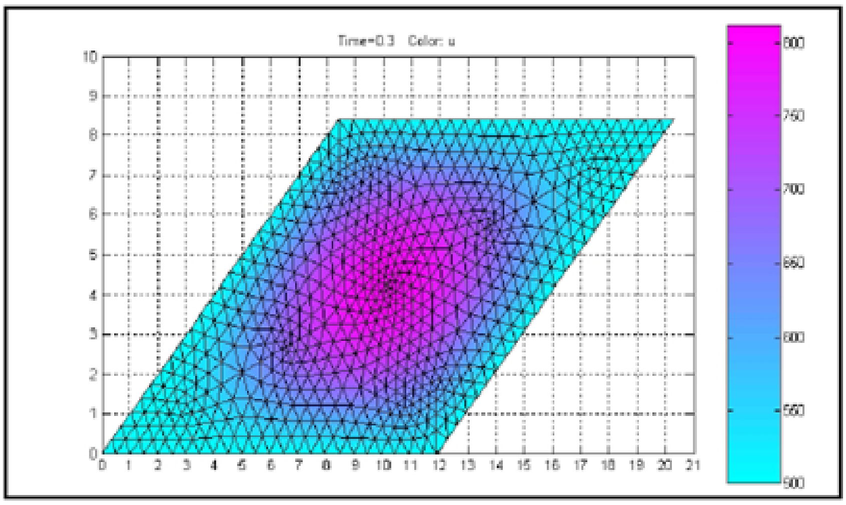


Figure 2.9: Fine-gridded matrix block for calculation of shape factors[93]

It is obvious that such an attempt would never get a chance to be industrially applicable. This is one of the main reasons that although the industry is aware of the limitations of the conventional transfer equation, they still widely apply it simply because *a practical alternative has not been offered yet.*

2.5 Permeability Tensor Measurement

Measuring the permeability tensor is a challenging task and many authors e.g.[5],[31],[59],[64],[88],[98],[100] have tried to design methods or instruments to measure it. In this section, the method suggested by Asadi et al.[5] is briefly explained.

A sample from the rock that is large enough to eliminate the end-effect is shaped as described by Asadi et al.[5] (which is the 3D extension of Rose[88] method for 2D):

The cubic sample is reshaped into a parallelepiped in three steps performed in each direction one after the other. The angle of each outlet face of the sample is changed to achieve the state in which the upper flow and the lower flow become the same (and the inlet face angle is also changed in parallel to the outlet face).

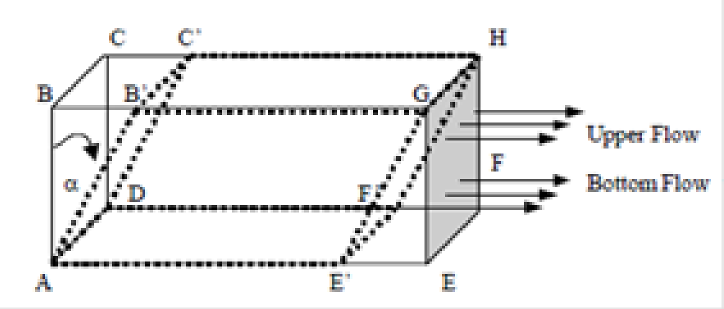


Figure 2.10: The first step for reshaping the cubic sample to measure anisotropy

Figure 2.10 shows the first step of reshaping the cube sample. The same procedure is also performed on the other two directions and ultimately the block is reshaped into a parallelepiped as shown in Figure 2.11. The plot of angle vs. $\ln\left(\dfrac{q_{top}}{q_{bottom}}\right)$ shall produce the final values for the angles of each face.

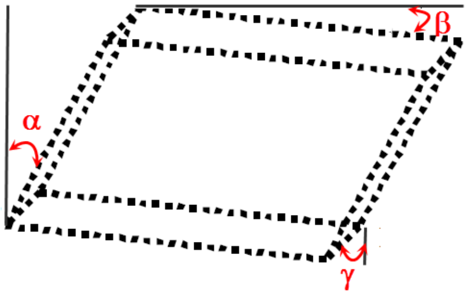


Figure 2.11: The final parallelepiped, formed by reshaping the cubic sample

It is expected that the direction of pressure gradient along each side of this parallelepiped is along the flow path.[5],[88] Now three permeability measurement experiments can be performed along each side of the parallelepiped measuring the pressure drop for a given flow rate and writing the Darcy's flow equation for that face as in Equation 2.18:

$$\vec{Q} = \frac{-\bar{\bar{k}}}{\mu}\nabla\vec{P} \Rightarrow \begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix} = \frac{-1}{\mu}\begin{pmatrix} k_{xx} & k_{xy} & k_{xz} \\ k_{yx} & k_{yy} & k_{yz} \\ k_{zx} & k_{zy} & k_{zz} \end{pmatrix}\begin{bmatrix} \partial p/\partial x \\ \partial p/\partial y \\ \partial p/\partial z \end{bmatrix}, \tag{2.18}$$

where the 9 permeability tensor values are unknown. Three additional equations come from the fact that the permeability tensor is symmetric by the Onsager[70] reciprocal relations:

$$k_{ij} = k_{ji}, \quad (i, j \in \{x, y, z\}) \ . \tag{2.19}$$

In order to calculate the remaining unknowns, Asadi et al.[5] consider the directions of maximum, minimum and intermediate permeability and their angles to the direction of pressure gradient: The angles of the maximum permeability direction to each component of the pressure gradient vector are M_1, M_2, M_3 respectively and for the minimum permeability direction, m_1, m_2, m_3 and for the intermediate permeability direction (which is perpendicular to both minimum and maximum permeability directions) are i_1, i_2, i_3. The principal permeabilities k_{xx}, k_{yy}, k_{zz} are the eigenvalues of the following matrix (Anton and Rorres[2]):

$$\begin{pmatrix} k_\lambda^{-1} & k_{xy} & k_{xz} \\ k_{yx} & k_\lambda^{-1} & k_{yz} \\ k_{zx} & k_{zy} & k_\lambda^{-1} \end{pmatrix}\begin{bmatrix} \cos M_1 \\ \cos M_2 \\ \cos M_3 \end{bmatrix} = 0, \tag{2.20}$$

$$\begin{pmatrix} k_\lambda^{-1} & k_{xy} & k_{xz} \\ k_{yx} & k_\lambda^{-1} & k_{yz} \\ k_{zx} & k_{zy} & k_\lambda^{-1} \end{pmatrix}\begin{bmatrix} \cos m_1 \\ \cos m_2 \\ \cos m_3 \end{bmatrix} = 0, \tag{2.21}$$

$$\begin{pmatrix} k_\lambda^{-1} & k_{xy} & k_{xz} \\ k_{yx} & k_\lambda^{-1} & k_{yz} \\ k_{zx} & k_{zy} & k_\lambda^{-1} \end{pmatrix}\begin{bmatrix} \cos i_1 \\ \cos i_2 \\ \cos i_3 \end{bmatrix} = 0, \tag{2.22}$$

$$\begin{aligned} &\cos M_1 + \cos M_2 + \cos M_3 = 0 \quad . \\ &\cos m_1 + \cos m_2 + \cos m_3 = 0 \\ &\cos i_1 + \cos i_2 + \cos i_3 = 0 \end{aligned} \tag{2.23}$$

Equation 2.18 to Equation 2.23 altogether contain 18 unknowns and 18 equations; by solving them, all the permeability tensor elements can be calculated as explained by Asadi et al.[5].

Chapter 3

The Elements of the New Approach

3.1 Introduction

In this chapter, some concepts, ideas and theories are presented in separate headlines which are independent from each other; however, all of the mentioned concepts in this chapter are related to the new presented approach of “simulating naturally fractured reservoirs using the recovery curves” which will be explained in details in Chapter 4.

The elements covered in the current chapter are all either devised, developed, improved or suggested *in this work* to justify the necessity of the new approach and/or to overcome the problems of creating and using recovery curves for dual-porosity simulation.

3.2 Shape Factor

The shape factor, as briefly introduced in Section 2.4.4, is used in the matrix-fracture transfer term (Equation 2.8) in the conventional dual-porosity model. In this part, some more details and noteworthy explanations about the shape factor are presented.

3.2.1 Pseudo-Steady State vs. Transient

In small-scale laboratory experiments and measurements, the fluid transfer between the matrix and the fracture is affected by the *transient* period, causing different results from when considering *pseudo-steady* behavior. However, as discussed and stated by Sarma and Aziz[94]:

> “The shape factor converges asymptotically to the pseudo-steady state shape factor for dimensionless times greater than 0.1, which in typical reservoirs, the real time equivalent to this is usually very small *(e.g. in the scale of a few hours to just a few days)* and it is often justified to only use the pseudo-steady state shape factor. The transient period can be significant in transient well tests, or in very tight gas reservoirs”.

This means that in practical field studies, where the time-steps are in the range of several days, the transient period would not have any influence on the results, and the pseudo-steady assumption accurately describes the conditions of the system (much less CPU-intensively). Therefore, in most industrial reservoir simulators, only pseudo-steady state shape factors, such as Kazemi-Gilman-Elsharkawy[52] (KGE), are used.

Moreover, the term "shape factor" semantically suggests that it is a *factor* based on the shape (geometry) of the system (and not the potential difference or other parameters). Therefore, it should be necessarily only dependent on the shape of the matrix block. However, some authors[e.g.[19],[75],[76],[86]] have considered the transient flow between matrix and fracture (which leads to shape factors that change in time) and used the term "*time-dependent shape factor*" which is logically incorrect; as the "shape" of the system does not change in time (and therefore, it is expected that the shape factor remains constant too). Note that the scale of time in which the behavior is transient, and these time-dependent shape factors are changing would be usually in a few hours to just few days[94].

The pseudo-steady state assumption, on the other hand, leads to a constant parameter for the geometry function i.e. the "shape factor". Since the goal of this work is simulation of the full-scale NFRs, only the pseudo-steady state geometry functions are considered. It is also noteworthy to mention that those time-dependent shape factors are not meant to account for the possible changes of the fracture network as a result of pressure change (such as fractures being closed), but the changes of the shape factor are only an attempt to consider the transient state effects.

3.2.2 Permeability Anisotropy

KGE[52] introduced a generalized shape factor formula for matrix blocks of any shape, with the pseudo-steady state and isotropic permeability assumptions. Heinemann and Mittermeir[48] mathematically *derived* the KGE shape factor formula for anisotropic cases, generalizing it for these blocks as well (as described in Appendix B). Unlike the pseudo-steady state assumption (which as discussed, does not have any practical effect in full field studies), the isotropic permeability assumption that many authors (like KGE[52]) have considered, can have a considerable effect on the full field behavior.

In order to demonstrate the significance of matrix permeability anisotropy on the matrix-fracture transfer, a piece of rock with the shape factor of $\sigma = 0.183\ m^{-2}$ and the effective height of $L_Z = 7.62\ m$ was modelled using the Single Matrix Block analysis under water imbibition drive (see Section 3.4, Appendix C and Appendix D). The apparent permeability was $k_a = 0.4$ and the principle permeability values (Section 2.4.1.1) are $k_x = 0.4,\ k_y = 0.12\ and\ k_z = 0.12\ mD$. The modeling was done once assuming the isotropic apparent permeability in all directions and once the actual permeability anisotropy was considered. The results of the isotropic and anisotropic cases (as well as their difference) is plotted in Figure 3.1.

As it can be observed in Figure 3.1, the two scenarios present two completely different trends of production: The isotropic case time vs. recovery factor curve starts with a higher slope than the anisotropic curve and this continues until about 40% of the oil is recovered, when the anisotropic curve's slope becomes higher than the isotropic one.

A closer look at the results, shows that recovering 40% of the matrix oil content, takes nearly 18000 days (~49 years) for the isotropic case while it only takes about 6000 days (~16 years) for the anisotropic case which is quite a remarkable difference of about 33 years in estimation of the production from the same block, with the same initial and boundary conditions.[4]

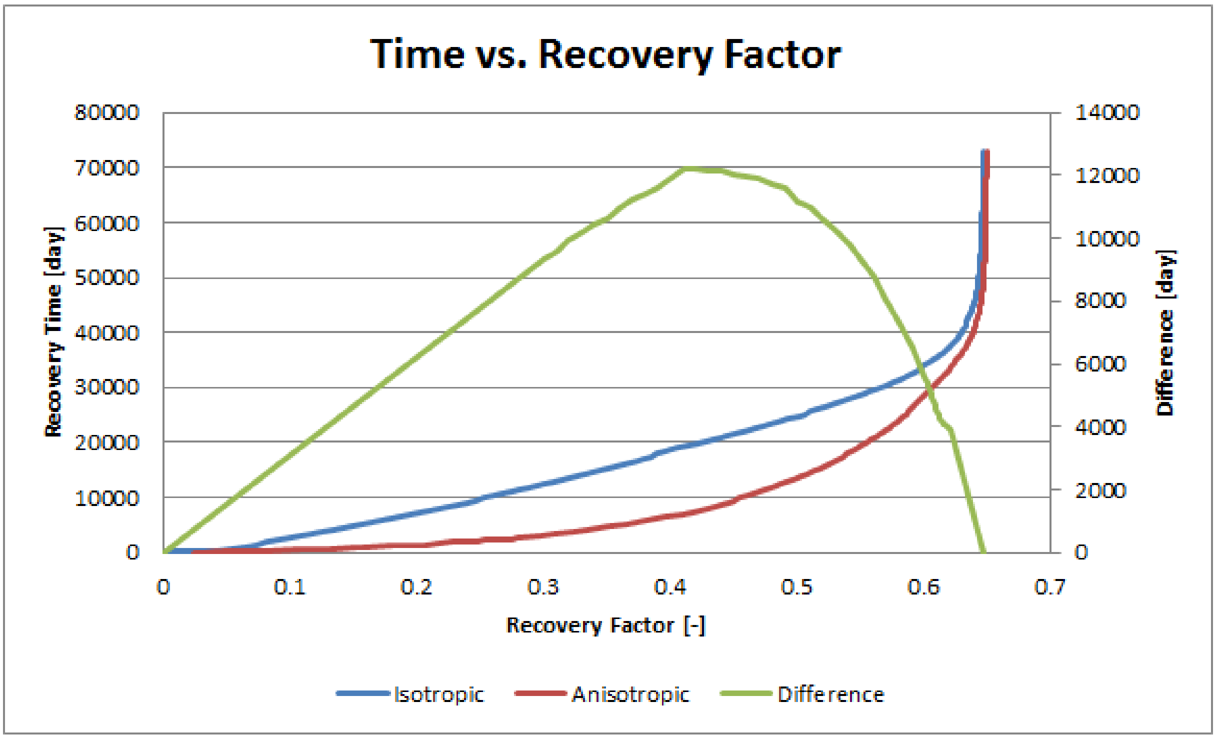


Figure 3.1: Isotropic & anisotropic time vs. recovery factor curves

It is also observable that the difference between the two curves starts on zero (obviously as the initial conditions are the same) and also ends on zero at 64.68% (the ultimate) recovery. The reason is that the ultimate recovery from the matrix block only depends on the driving mechanisms in action and the saturation endpoints which are the same in both scenarios and is not dependent on the permeability or viscosity. In other words, the two scenarios will ultimately produce the same amount of oil but with different trends. However, it is essential that this trend is known as accurately as possible to be able to make appropriate plans for the production from naturally fractured reservoirs.

Figure 3.1 demonstrates the significant effect that permeability anisotropy, which is commonly ignored in the industry, can have on the matrix-fracture transfer; therefore, in this work, a rather simple method to measure the anisotropic permeability tensor on outcrops of reservoir formations is proposed (Section 3.2.2) and it is suggested to be practiced and used in reservoir simulations along with shape factor formulas which take the anisotropy into account (such as Heinemann-Mittermeir[48]).

3.2.3 Shape vs. Shape Factor

The shape factor is generally based on the surface-to-volume ratio of the model as introduced by Barker[9]:

$$\sigma = \alpha\left(\frac{V_m}{A_m}\right)^{-2}, \tag{3.1}$$

where V_m is the volume, A_m is the surface of the matrix block and α is a dimensionless parameter. It means that the flow behavior of the model depends not really on the shape, but also on its *surface-to-volume ratio*. Consequently, if the surface-to-volume ratios of two different matrix blocks of different shapes are the same, the matrix-fracture flow will be the same as long as the driving forces are the same. The considered driving forces are capillary imbibition (between the fracture and the matrix) and gravity drainage (which depends on the height of the matrix block).

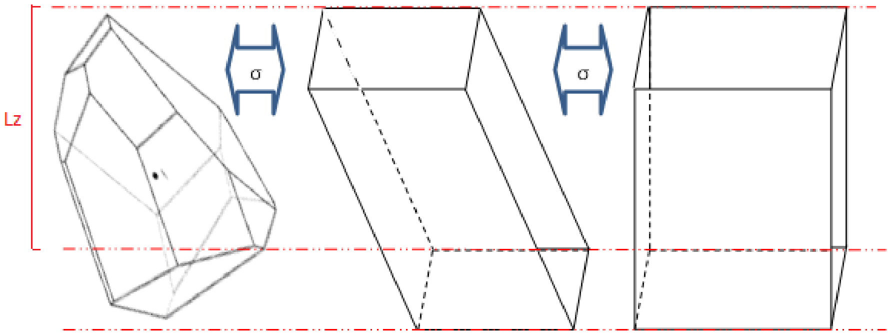


Figure 3.2: Different matrix block IDs (shapes) with the same shape factor and flow behavior

Therefore, if the heights of the blocks are the same (to impose the same gravitational drive), all models will have the same shape factor, as long as their surface-to-volume ratio remains the same. It is illustrated in Figure 3.2 how an arbitrarily shaped matrix block, can have the same shape factor as a cuboid with the same height. This means that in order to model the flow behavior of a matrix block of any shape, it is possible to use a simple cuboid (which is easier to model and simulate) with the same “shape factor” as the irregularly shaped matrix block. The procedure is explained in Section 3.3.6.

3.2.4 Shape Factor Is Not a History-Matching Parameter

The shape factor is a meaningful physical parameter that reflects the geometry of the matrix block; and can be theoretically derived from the flow equations for pseudo-steady state conditions as shown by Heinemann-Mittermeir[48] and explained Appendix B.

This means that the shape factor is in a similar category of properties such as porosity and is not just a meaningless multiplier of the matrix-fracture transfer equation that can be regarded as a simple history-matching parameter. It should be tried to estimate the shape factor as accurately as possible and regard it as a statistical parameter. Therefore, the statement: "*The shape factor acts as a multiplier on the matrix-fracture coupling, it may simply be treated as a history matching parameter.*"[95] from ECLIPSE simulator technical description is not correct and can lead to completely unreliable history-matched models that cannot be trusted for predictions.

3.3 Measuring the Shape Factor

The parameters and steps necessary to use the pseudo-steady state shape factor after Heinemann-Mittermeir[48] (HM) for an irregularly shaped anisotropic piece of rock, are described in this section:

3.3.1 Global Coordinate System

Consider a homogeneous anisotropic matrix block of any shape. A global 3-Dimensional Cartesian coordinate system is required to make the measurements. The origin can be any arbitrary point in space (based on which all the distances will be measured). Such an arbitrary global coordinate system and a piece of rock are shown in Figure 3.3.

3.3.2 Surface Area Measurement

In geometry, Heron's formula named after Heron of Alexandria, states that the area T of a triangle with side-lengths of a, b and c (as described by Dunham[30]) is:

$$\begin{aligned} T &= \sqrt{s(s-a)(s-b)(s-c)} \quad . \\ s &= (a+b+c)/2 \end{aligned} \tag{3.2}$$

This formula can be used to measure the surface of any irregular shape by masking it with non-overlapping triangles (as shown in Figure 3.4), measuring their sides, calculating their areas and adding them up for all the triangles to calculate the surface area of the object.

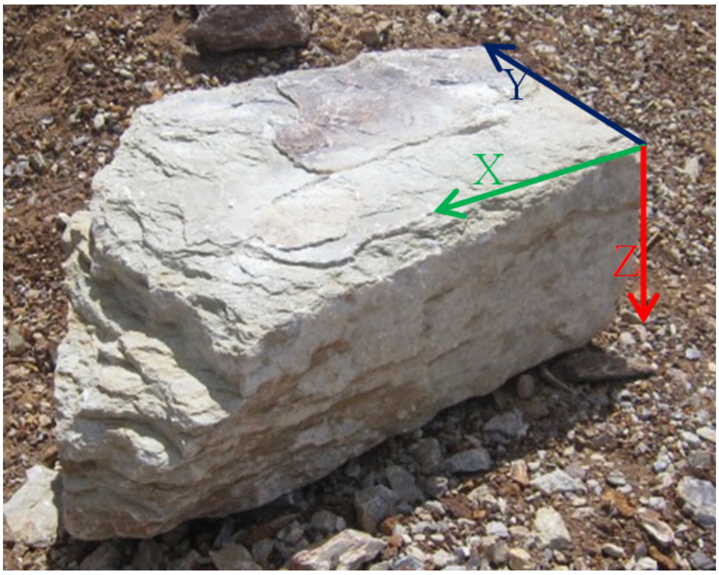


Figure 3.3: Matrix block; Arbitrary global Cartesian coordinate system

3.3.3 Measuring the Volume of the block

The corner points of Figure 3.4 can be used to divide the irregular shape to 3D polygons (such as pyramids) and calculate the total volume of the block (V_m) based on the volumes of these polygons.

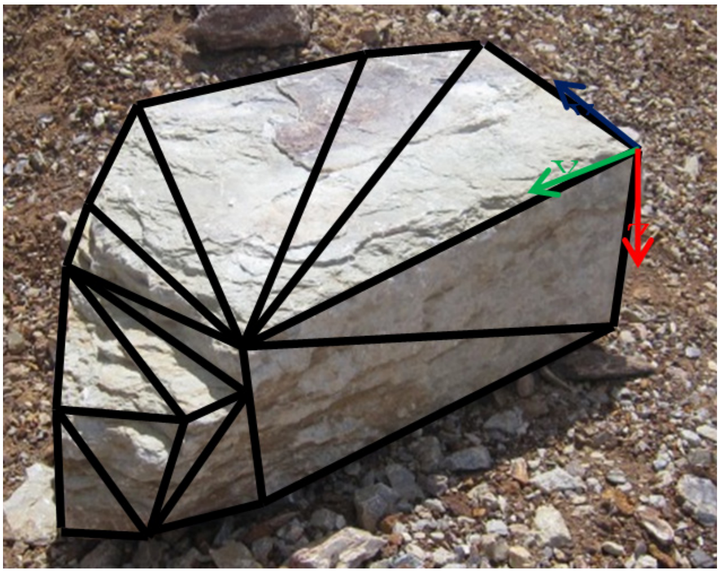


Figure 3.4: Measuring the surface area and the volume

3.3.4 Finding the centroid of the block

The centroid of the matrix block shape can be calculated from the positions of the corner points:

$$\vec{r}_c = \frac{1}{n}\sum_i^n \vec{r}_i\ , \tag{3.3}$$

where $\vec{r}_c$ is the position vector of the centroid, n is the total number of corner points and $\vec{r}_i$ is the position vector of each corner point of the block (shown in Figure 3.4).

3.3.5 Face distance from the centroid

The distance of a surface to a point can be calculated[101] from Equation 3.4:

$$d = \hat{n} \cdot (\vec{r}_O - \vec{r}_c), \tag{3.4}$$

where d is the distance of the surface to the centroid, $\hat{n}$ is the unit normal vector of the surface, $\vec{r}_O$ is the position vector of any arbitrary point on the surface and $\vec{r}_c$ is the position vector of the centroid of the block.

3.3.6 Calculating the Representative Cuboid

The representative cuboid which has the same shape factor as the irregularly shaped block can be calculated assuming $L_x = L_y$, and L_z = *the effective height of the original block* (to keep the effects of gravity which is height-dependent the same between the cuboid and the original rock, see also Figure 3.2). The values can be easily calculated for a cuboid from Equation 3.14 for KGE shape factor or Equation 3.15 for HM shape factor as explained in Section 3.4.7.

3.3.7 Finding unit normal vectors of each block face

Three arbitrary points are chosen on the desired surface: $O(o_x, o_y, o_z)$, $A(a_x, a_y, a_z)$, $B(b_x, b_y, b_z)$ as shown in Figure 3.5. The unit normal vector $\hat{n}$ can be calculated from the cross product of the two vectors of $\overrightarrow{OA}$ and $\overrightarrow{OB}$, divided by its length as in Equation 3.5:

$$\hat{n} = \frac{\overrightarrow{OA} \times \overrightarrow{OB}}{\left|\overrightarrow{OA} \times \overrightarrow{OB}\right|} \tag{3.5}$$

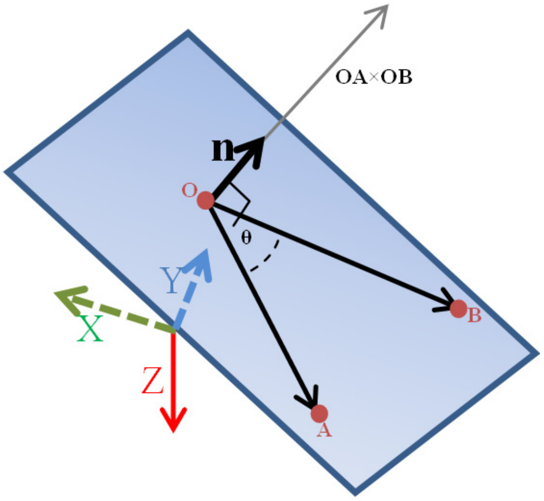


Figure 3.5: Finding the unit normal vector of the face

3.3.8 Principle permeability coordinates and permeability tensor diagonalization

The permeability tensor can be measured by any feasible method; for instance the method which was explained in Section 2.5.

Since the permeability tensor is[49] both symmetric and positive definite(1), it is always diagonalizable[2]. In order to diagonalize the permeability tensor, its eigenvectors, $\overrightarrow{X}_1$, $\overrightarrow{X}_2$, $\overrightarrow{X}_3$, respectively for the eigenvalues of k_x, k_y and k_z should be calculated. Equation 3.6 is the definition of eigenvalues and eigenvectors for the permeability tensor:

$$\overline{\overline{k}}'\overrightarrow{X}_1 = k_x\overrightarrow{X}_1,\ \overline{\overline{k}}'\overrightarrow{X}_2 = k_y\overrightarrow{X}_2,\ \overline{\overline{k}}'\overrightarrow{X}_3 = k_z\overrightarrow{X}_3, \tag{3.6}$$

$$E = \begin{bmatrix} \overrightarrow{X}_1 & \overrightarrow{X}_2 & \overrightarrow{X}_3 \end{bmatrix} = \begin{bmatrix} X_{1_x} & X_{2_x} & X_{3_x} \\ X_{1_y} & X_{2_y} & X_{3_y} \\ X_{1_z} & X_{2_z} & X_{3_z} \end{bmatrix}, \tag{3.7}$$

where E is the matrix comprised of eigenvectors of $\overline{\overline{k}}'$ written as column vectors and put side by side.

(1) Positive definite matrix: a matrix $\mathbf{M}$ such that $\mathbf{x}^T\mathbf{M}\mathbf{x} > 0$ for all nonzero $\mathbf{x}$.

Since $\overline{\overline{k'}}$ is positive definite, E is an orthonormal[1] matrix and represents simply a rotation transform.[2] Because k_x, k_y, k_z are the eigenvalues of the permeability tensor, Equation 3.8 and as a result, Equation 3.9 are always true (by definition)[2]:

$$\overline{\overline{k'}}E = E \cdot \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_z \end{bmatrix} \Rightarrow \overline{\overline{k'}} = E \cdot \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_z \end{bmatrix} \cdot E^{-1}, \tag{3.8}$$

$$\overline{\overline{k'}} = E \cdot \overline{\overline{k}} \cdot E^{-1}. \tag{3.9}$$

$\overline{\overline{k}} = \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_z \end{bmatrix}$ is the *diagonalizedpermeability tensor* in the principle permeability direction.

This expression helps describe the procedure of matrix diagonalization in the following sequence of operations (reading Equation 3.9 from the right):

1. First, multiplying (the space) by E^{-1}. It can be thought of as performing a linear change of coordinates, i.e. transforming the coordinates to some *special* coordinate system called the “*principle permeability coordinate system*”.
2. In that coordinate system, multiplying by $\overline{\overline{k}}$. But this is a particularly easy matrix to multiply with, since the coordinates do not mix: it means that each coordinate gets stretched or squeezed by just one of the eigenvalues of $\overline{\overline{k'}}$.
3. Finally, going back to the global coordinates, multiplying by the inverse of E^{-1}, namely E.

Note that the order of choosing the eigenvalues as k_x, k_y or k_z is not important, since the rotation transform matrix is created by the corresponding eigenvectors in the same order as the selected eigenvalues. In other words, the values of the diagonalized permeability tensor in the transformed coordinate system, is always the same, regardless of the order of selection of the eigenvalues as x, y and z subscripts.

Applying the rotation transform of E^{-1} to the global coordinate system, the principle permeability coordinate system (X', Y', Z') can be calculated as shown in Figure 3.6 with the dashed arrows. In this coordinate system, the diagonalized permeability tensor can be directly used in calculations.

[1] Orthonormal matrix: matrix $\mathbf{M}$ where $\mathbf{M}^T\mathbf{M} = \mathbf{I}$.

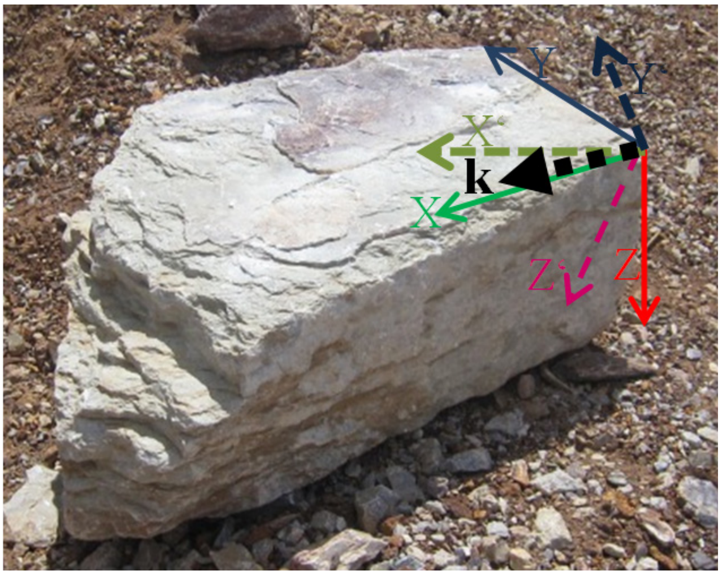


Figure 3.6: Principle permeability coordinate system

3.3.9 Equivalent isotropic permeability and normalized anisotropy tensor

Muskat[65] introduced the term *equivalent isotropic permeability* to calculate the isotropic permeability in case of having anisotropic permeability tensor (to be used in calculations and simulations that are based on the isotropy assumption) which is nothing more than the geometric mean of the directional permeability values:

$$k_a = \sqrt[3]{k_x k_y k_z} \ . \tag{3.10}$$

This value can be used to normalize the permeability tensor to a dimensionless tensor that represents the anisotropy of the medium which is called dimensionless normalized anisotropy tensor, which represents the anisotropy of the rock:

$$k'_p = \begin{bmatrix} k'_x & 0 & 0 \\ 0 & k'_y & 0 \\ 0 & 0 & k'_z \end{bmatrix} = \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_z \end{bmatrix} / k_a . \tag{3.11}$$

3.3.10 Practical Simplifications

The presented method to measure the shape factor parameters from outcrops, is mathematically accurate, but would be difficult to manually practice on a large scale. However, there are software applications (e.g. *3D Software Object Modeller Pro*[22]) that help make 3-Dimensional models from the pieces of rock just by analyzing pictures from different angles of the rock. These 3D models can be then used to easily determine the surface areas, centroid and the distance of each face to the centroid and all the other necessary parameters required to ultimately calculate the shape factor using Equation 2.10.

An alternative practical method is to express the Heinemann and Mittermeir[48] shape factor formula, as in Equation 3.12, in a form which facilitates estimating the parameters intuitively:

$$\sigma^{HM} \approx \frac{1}{V_m}\left(\frac{A_{K_{min}}}{d_{K_{min}}} + \frac{A_{K_{max}}}{d_{K_{max}}} + \frac{A_{K_{orth}}}{d_{K_{orth}}}\right), \tag{3.12}$$

where V_m is the volume of the matrix block, K_{min}, K_{min} and K_{orth} are the inspected directions of minimum and maximum permeability and a direction orthogonal to both the latter, respectively. A_K is the projected area of the matrix block on a plane orthogonal to the direction of each $\vec{K}$ and d_K is half of the mean length of the matrix block in the same direction. Figure 3.7 shows the schematic representation of A_{Kmax}, the projected area and d_{Kmax}, half of the mean length of the matrix block in the direction of maximum permeability.

It is possible to estimate A_K and d_K as well as the matrix block's volume, V_m, by some measurements intuitively which makes utilizing this shape factor more practical onsite. These principle permeability *directions* can be estimated itself by close inspection of the formation sedimentation. The permeability *values* can be measured in lab for some cores and averaged for the formation.

Note that in Equation 3.12, the *normalized* directional permeability values are not explicitly observable; as their effects are already included in the terms A_K/d_K (i.e. since this terms are determined in the directions of principle permeability, they are dependent on the anisotropic permeability tensor). However, the (average) apparent permeability, k_a, has to be measured in the laboratory at least for some samples from the formation.

Such simplifications make it practical to use outcrop investigations to measure the anisotropic shape factor more accurately than what is currently being used, but still not as time-consuming and accurate as measuring everything in the lab for the pieces of rock, as described earlier (which could make this procedure impractical or even impossible in large scale).

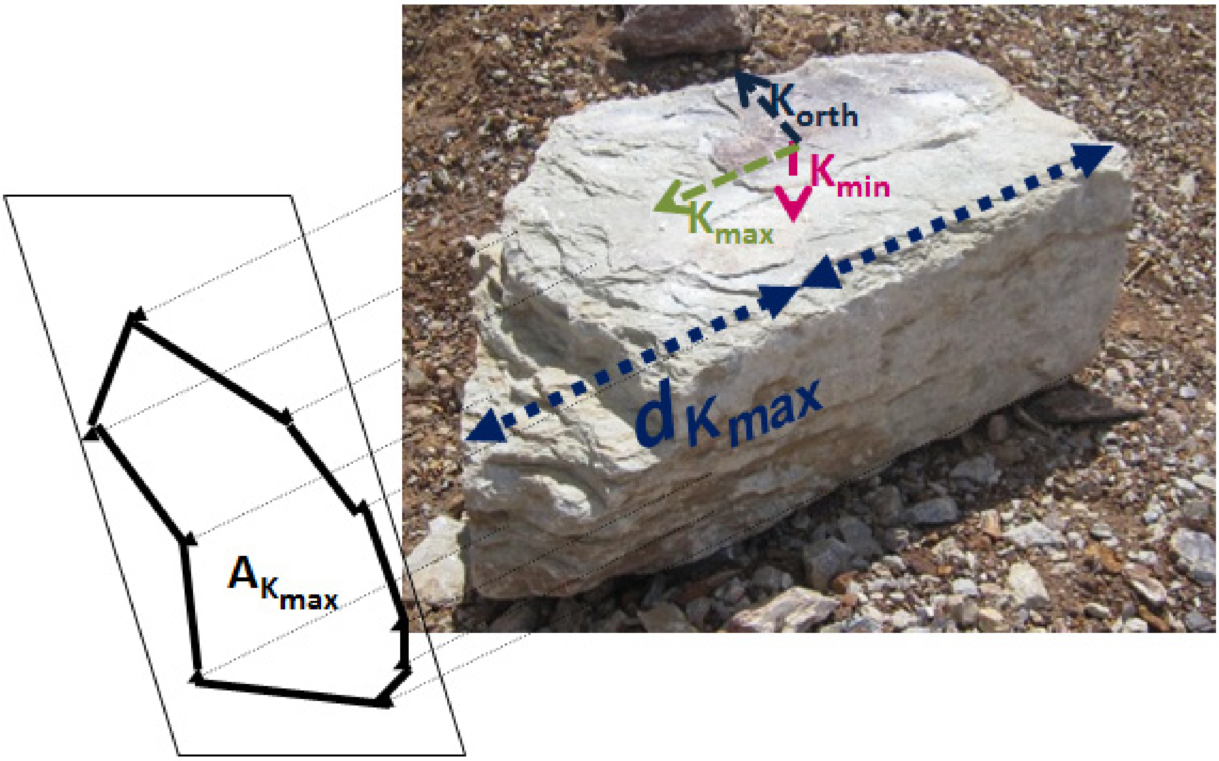


Figure 3.7: Area and length of the matrix block in the principle permeability direction

3.4 Single Matrix Block Analysis

The idea of using Single Matrix Block (SMB) models comes from the ambition to simulate laboratory experiments of studying the matrix block depletion under different conditions and the recovery processes for every single *matrix block* (Section 4.3.1) and to generate the recovery curve under the specified conditions. Such an approach (and laboratory experiment) gives a detailed insight about the processes that are going on between the matrix and the surrounding fractures at different places and under different conditions.

This would be the most general and the best approach, if it could be practiced throughout the reservoirs. But unfortunately, this approach is not applicable on the reservoir scale, both because it is impossible to access different *matrix block*s (in full) from all around the reservoir to take them to the lab, and that such experiments needs a lot of time and cost to perform and produce the desired outcome. Therefore, SMB laboratory experiments remain restricted only to some research studies and not a standard practice in reservoir modeling.

In general, the experiment (explained in Appendix C in details) is to submerge the matrix block in a fluid and collecting the extracted oil and plotting the recovered oil volume versus time. This plot is called a "*Recovery Curve*" which demonstrates how the matrix block under study, depletes in time with the ceratin boundary conditions and active driving mechanisms of the experiment (such curves are shown in Figure 3.22, Figure 4.7 and Figure 4.9 for different *matrix blocks* and driving mechanisms). As will be explained in Section 4.3 in details, knowing the *matrix block* distribution within a given simulation cell, it is possible to accurately estimate the overall recovery curve for the simulation cell.

As stated earlier, since the actual laboratory experiments are not practical (enough) to be used routinely for the real reservoirs, single matrix block models have been developed to numerically simulate such laboratory experiments which makes the study much easier and more affordable. Different applications of small-scale single matrix block models can be found in the literature[e.g.[20],[33],[41],[54],[89]]. These publications deal with recovery mechanisms in fractured reservoirs, primarily with water/oil countercurrent imbibition and gas/oil drainage, and the matrix-fracture transfer.

Most of these calculations have been made using commercial simulators placing production and injection wells in the fracture cells. However, such a workaround (to model the laboratory experiment) is impractical and in most cases numerically instable, and will produce questionable results with long CPU times. Some of the problems of using commercial simulators to model single matrix blocks are mentioned:

- For every matrix block, a new grid model has to be created which requires recreating the grid cells, setting permeability, porosity distributions etc. This is undoubtedly a tedious work and one of the main reasons why such an informative analysis is not even numerically practiced in full field studies.
- The very small pore volumes of the grid cells, especially for the fracture cells causes numerical instability.
- The necessity to use bottomhole-pressure-controlled production (and/or injection) wells to maintain the fracture conditions causes further numerical instability.
- The boundary conditions for the matrix are determined by the fracture pressure mainly since the fracture saturation cannot be even set as the boundary condition; and even so, these parameters cannot be properly set in such a conventional grid model.
- Due to the matrix-fracture fluid transfer, the fluid contents of the fracture changes in time. Assuring constant pressure and saturations in the fracture cells (as in the laboratory experiment of Appendix C) is required for a proper SMB investigation. Setting both pressure and saturations of the fracture cells as constant is in no way achievable in commercial reservoir simulators.
- The influence of viscous displacement cannot be considered in such models because no fluid movement can be incorporated along the fracture cells.

Pirker[81] made some experiments with some commercial software tools, mostly with ECLIPSE[95] and showed that a more sophisticated approach than to model each single matrix block separately is necessary. Pirker[81] tried to standardize this method and developed it to some extent leaving it open with some suggestions.

In this work, the single matrix block analysis concept suggested by Pirker[81] is completely developed and integrated in ***PRS***[45] (described in Appendix D), which is available to be used in any reservoir model for different *matrix block*s and conditions with minimal effort and can also be implemented in any conventional reservoir simulator. The objective of this section is to introduce a practical and efficient method for performing Single Matrix block calculations. This method should satisfy the following requirements:

- The calculation should be mathematically and numerically correct; e.g. using fully implicit control volume finite difference method on a fine orthogonal grid. The result will be the oil recovery as function of time or dimensionless time.

- The recovery mechanism of a single matrix block should be calculated without any simplifications, by considering all driving forces, pressure and saturation distribution within the matrix and changing states in the surrounding fractures.

- Only black oil models are considered.

3.4.1 New Features and Improvements

In this work, the Single Matrix Block Analysis concept after Pirker[81] was modified, extended and new features were added to complete it. Here only the differences are briefly mentioned:

1. Pirker, after many studies on different SMB models, advised that the simulation grid and run setup had to be standardized and optimized for a proper use of the on single matrix block models. This advise is followed; however, three more layers have been added (for the reasons described in Section 3.4.3).
2. Pirker implemented the SMB analysis only for water imbibition and gravitational drives, based on one simple example. This is now implemented as a standard option in ***PRS***[45] commercial simulator and is readily usable for *any* reservoir model.
3. Pirker did not implement the gas gravity drainage. The main reason could be that for the gas drive case, in such small models, the convergence is even poorer than water drive is. The gas gravity drainage drive was successfully implemented, standardized and tested like the water drive and is available in ***PRS***[45].
4. The time-step regulation was completely revised and redesigned in order to allow better convergence for the small cells.
5. The problem of imposing both a constant pressure and a constant saturation (at the beginning of each time step) for the fractures was successfully overcome and therefore, much better convergence is achieved with results that are completely physical (no oscillations nor negative second derivative for the recovery curve are now observed, as they are not theoretically excepted either).
6. The single matrix block analysis recovery curves are output versus both the normal time (in days) and the dimensionless time, so that each one can be used properly wherever needed.
7. A completely new option to allow adding a pressure gradient for considering the effect of *viscous displacement* in the SMB analysis is devised and implemented.

8. Pirker implementation only considered two vertical fracture sets, but used a shape factor formula for a case that is surrounded by fractures (i.e. also with horizontal fractures) which was a mistake in Pirker's implementation. In this work, by default the matrix block is considered to be surrounded on all sides by fractures (i.e. both vertical and horizontal fractures) and also optionally allows sealing the block on the top and bottom faces to consider only the vertical fractures (see Figure 3.8). For each case, unlike in Pirker's implementation, the correct shape factor formula will be used.
9. Pirker imagined that using a dimensionless time for the recovery curve would help, but did not explain why and how; therefore, only used the normal time for the actual recovery curve calculation (even though Pirker had introduced an empirical dimensionless time formula). In this work the idea of using the dimensionless time is proven to be not only helpful but necessary. Additionally, from the flow equation a proper dimensionless time for this purpose is *derived* (Equation 3.26) and also its applications and benefits are explained (Section 4.3.2 and Section 5.2.2.1).
10. The SMB Analysis is made generally available in a commercial reservoir simulator (Appendix D) for all reservoir models with just a single keyword in the input for many different conditions. A graphical user interface tool (Section D.4) is also developed which allows creating such curves, putting them in a database (for each reservoir model) and recall them for future use along with the functionality to lump them (Section 3.9) and for preparing the recovery curve regions (Section 5.2.1.4.1).

Note that for a matrix block, the *z*-direction in this case is not necessarily vertical, however, it should be considered that all parameters in the single matrix block examination are highly hypothetical. Therefore, in order to keep it simple, it is advisable to have the model vertical and assume that the permeability coordinate axes are identical with the geometrical ones.

3.4.2 The SMB Grid

In order to get reliable results, the SMB should represent credible and scalable real matrix blocks. In this respect, not the volume nor the average permeability is decisive, but the shape factor and the effective pressure gradient along the matrix block sides are. This is so because the recovery processes are, in fact, only defined by the saturation end-points and the acting deriving mechanisms. The volume and permeability play as scaling factors on the recovery and depletion time respectively.

The shape factor depends on the matrix-fracture connection surfaces, their distance from the center of gravity, the orientation of principle permeability axes and the surface-to-volume ratio explained in Section 3.2.3 and derived in Appendix B.

One conclusion of Heinemann-Mittermeir[48] mathematically derived shape factor, is that any arbitrarily shaped matrix block can be mapped to any other arbitrary shape, as long as the surface-to-volume ratio, by considering the permeability anisotropy, are identical. In other words, it is possible to find many geometries that have the same shape factor (and therefore the same flow behavior).

So once the shape factor for the actual (irregularly shaped) matrix block is measured, one can calculate a vertical cuboid which behaves identically to the actual irregularly shaped matrix block (as long as the driving forces are the same).

For instance the resulting recovery curve on the three geometries shown in Figure 3.2 have to be the same if their shape factors are equal. The third model is easy to discretize correctly with orthogonal Cartesian grid but for the other ones would be more time-consuming and difficult (while the results will be approximately the same for all cases). Therefore, it is of course preferred to use of the cubic model to investigate the matrix behavior numerically.

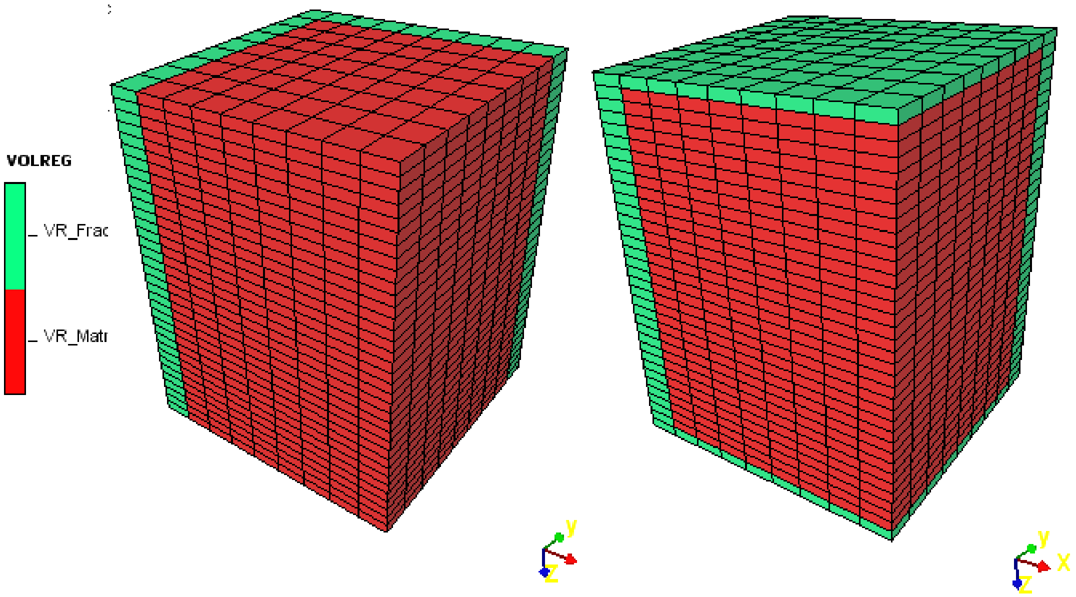


Figure 3.8: Single matrix blocks with different fracture constellations (fracture cells in green and matrix cells in red). On the left: two vertical fracture sets along *x*- and *y*-planes. On the right: two vertical fractures plus a horizontal fracture set on top and bottom.

There are two basic methods to construct the Cartesian grid: One using the matrix width, length and the effective height, and the other one using the shape factor and the effective matrix block height as briefly explained here (and in Section 3.4.7 in more details):

1. The vertical cuboid's size can be calculated by specifying the shape factor, σ, for which the investigation should be performed, the anisotropic permeability values (k'_x, k'_y, k'_z and k_a from Equation 3.11), and the effective matrix block height in the direction of gravitational force, L_z, using Equation 3.13:

$$L_x = L_y = \sqrt{(k'_x + k'_y)/(\sigma/4 - k'_z/L_z^2)} \; . \qquad (3.13)$$

2. The vertical cuboid can also be created by specifying the matrix block side lengths, L_x, L_y and L_z (note that L_z is here also the effective matrix block height in the direction of the gravitational force). The shape factor can (optionally) be calculated from Equation 3.14 but is not used in the run.

As discussed, the results from a regular cuboid for which the *X, Y, Z* coordinate axes are the principal permeability directions can be assumed to be also valid for any matrix block having the same shape factor and driving forces. Note that the *z*-direction in this case is not necessarily vertical. It should be seen that all parameters in the single matrix examination are highly hypothetical.

The matrix block model is discretized into grid cells. Fracture planes are added to the sides of the matrix block. The simple geometrical shapes of the cubic single matrix block models do not require any special gridding technique, therefore simple Cartesian grids are used to discretize the matrix block and the surrounding fracture planes.These fracture planes can be set up in different combinations. They can surround the entire matrix block or only the (vertical) sides. Figure 3.8 shows a quarter of the model for the two patterns of possible fracture constellations.

3.4.3 Model Dimensions

Due to the symmetry of the cubic block model, it is sufficient to conduct the simulation only on one quarter of the model, leading to considerable calculation time reduction. Since only a quarter of the cube is calculated, the side length of the cube is reduced in *x*- and *y*-direction to $L_x/2$ and $L_y/2$ respectively. The dimension in *z*-direction remains the same as in the vertical direction the gravity is in action making it not feasible to halve the block height. Figure 3.8 shows a single matrix block model, as used in the further investigations with two- and three-dimensional fracture networks (*2D* or *3D*).

Pirker[81] conducted a sensitivity analysis of the effect of the grid resolution on numerical accuracy and calculation time. Different grid resolution scenarios were calculated on the quarter model.

Figure 3.9 shows the results with *4*x*4*x*18*, *8*x*8*x*36* and *16*x*16*x*72* matrix grid cells. With the vertical fracture planes the grid models have dimensions of *5*x*5*x*18*, *9*x*9*x*36* and *17*x*17*x*72* grid cells respectively. Based on a greater number of numerical experiences, Pirker[81] recommends the *9*x*9*x*36* grid cells for the quarter model as generally applicable.

In this work, one layer was added to have a mid-layer based on which the initial equilibrium properties are calculated which makes the model dimensions *9*x*9*x*37*. This change was tested against the Pirker[81]'s *36*-layer case and the difference was negligible. In order to also consider the horizontal fractures, one additional fracture layer is added to both top and bottom of the model which will be deactivated if a *2D* fracture network (i.e. only vertical fractures) is considered. This makes the final grid dimension to *9*x*9*x*39* for the single matrix block model including both the matrix and fracture cells.

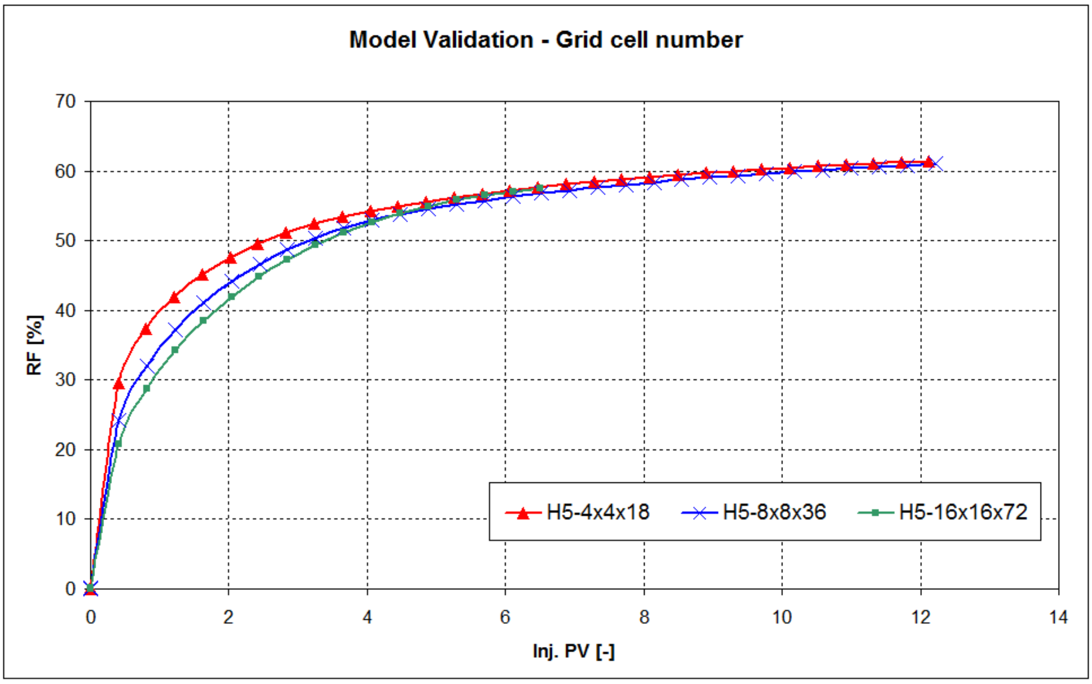


Figure 3.9: Sensitivity analysis on grid cell number (after Pirker[81])

3.4.4 Grid Cell Properties and Parameters

There are two different (fluid-in-place) volume regions: VR_Frac (green) for the fracture (single-porosity) domain and VR_Matr (red) for the matrix (single-porosity) domain as shown in Figure 3.8. The actual volumetric calculations will be performed based on the matrix volume region as only this is the representative of the matrix block under study. The fracture domain is modeled just as the boundary to the matrix domain.

The matrix domain is considered to be, homogeneous (as it is representative of a single matrix block, i.e., one purple polygon in Figure 2.6). The porosity of all matrix cells are, therefore, the same as that of the actual matrix block. The net-to-gross ratio is considered to be *1.0* for obvious reasons for the whole model.

The fracture domain's volume has no effect in the process as that domain's content is not considered in the calculations. However, it has to be large enough, so that the saturation of the fracture cells does not effectively change during each time step while oil flows out from the matrix until it is taken out at the end of the time step.

The matrix domain inherits the rock data of the original matrix block. The single-porosity fracture domain, has straight-line relative permeabilities and no capillary pressure. Note that the fracture domain here is not intended to be a medium of fluid flow, but just a boundary to the matrix cells. There is no theoretical restrictions on the PVT data, except that the PVT data for both the matrix and the fracture domains are essentially the same.

3.4.5 Transmissibilities

There are three types of inter-cell transmissibilities that can be considered:

1. *matrix cell to matrix cell*, this is the single-porosity transmissibility between the matrix cells (the red cells in Figure 2.6). The values of k'_x, k'_y, k'_z are used to calculate the transmissibilities in *x*-, *y*- and *z*-directions respectively. Note that since all cells are of the same size (and with the same parameters), the value of transmissibility in each direction is the same for all matrix cell pairs.
2. *fracture cell to fracture cell*, since the flow in the fracture domain is uninteresting (and the fracture domain content will be always restored to the initial state as will be explained in Section 3.4.8), the fracture-fracture transmissibility is by default set to zero to prevent the uninteresting flow between the fracture cells.
3. *matrix cell to fracture cell*; the boundary matrix cells can have one, two or three fracture neighbors. The transmissibility is calculated as in the single porosity model but only considering the matrix cell length and permeability (as the fracture cell acts just a boundary to the matrix cells).

3.4.6 Permeability Anisotropy

The single matrix block can have different permeability values in *x*-, *y*- or *z*-directions. Since it is assumed that the grid is in the principle permeability coordinate system (Section 3.3.8), the principle permeability values of k_x, k_y and k_z can be assigned in each coordinate direction. The normalized dimensionless permeability anisotropy values are $k'_x = k_x/k_a$, $k'_y = k_y/k_a$ and $k'_z = k_z/k_a$ in *x*-, *y*- and *z*-direction in the principle Cartesian coordinate system.

3.4.7 Shape Factor

One may argue, in Figure 3.8, the shape factors are different for the two cases (which is a correct argument). Note however, that the shape factor cannot be even defined in the single matrix block model as it is a single porosity matrix block (i.e. a single piece of rock) and should not be confused with the conventional dual-porosity model. Here, the shape factor is used to determine the geometry and size of the matrix block model (as described in Section 3.4.8) as well as for outputting purposes and for further analysis. It is never used in the course of calculations in the single matrix block model.

This means that when creating the model with *2D* or *3D* fracture network (i.e. without and with horizontal fracture planes respectively), the matrix block size is calculated from the correct simplification of Equation 2.10 for a cuboid. For the *3D* case Equation 3.14 and for the *2D* case Equation 3.15:

$$\sigma = 4\left(\frac{k'_x}{L_x^2} + \frac{k'_y}{L_y^2} + \frac{k'_z}{L_z^2}\right) \Rightarrow L_x = L_y = \sqrt{\frac{k'_x + k'_y}{\frac{\sigma}{4} - \frac{k'_x}{L_z^2}}}\ , \tag{3.14}$$

$$\sigma = 4\left(\frac{k'_x}{L_x^2} + \frac{k'_y}{L_y^2}\right) \Rightarrow L_x = L_y = \sqrt{\frac{k'_x + k'_y}{\frac{\sigma}{4}}}\ . \tag{3.15}$$

In this way, the effect of different fracture constellations will be considered in the matrix block size through usage of different shape factor formulas. Note that if the shape factor after Kazemi *et al.*[52] is to be used instead of Heinemann and Mittermeir[48] shape factor, the values of k'_x, k'_y, k'_z would be set equal to *1* in Equation 3.14 and Equation 3.15 as the permeability in KGE[52] shape factor is considered to be isotropic.

3.4.8 SMB Model Initialization, Drive Mechanisms and Production Method

Pirker[81] showed that using bottomhole-pressure-controlled wells to handle injection into and/or production from the fracture domain is instable, time-consuming and impractical; and suggested using fractures as boundaries to the matrix domain. In doing so, the fracture content should not change in time and the oil that flows from the matrix domain to the fracture domain, is extracted when it enters the fracture domain so that the fracture domain content always remains as it originally was. This condition is also valid in the actual laboratory experiment of placing the matrix rock in water (or gas) as the extracted oil would immediately rise up (or fall down) due to buoyancy, leaving the matrix again completely surrounded by*100%* water (or gas) as explained in Appendix C. Therefore, the appropriate boundary conditions are the fluid saturations and not the phase potentials for the single matrix block model.

The amount of extracted oil from the fracture domain, which is the same as the entered amount of water (or gas) into the matrix domain, is directly collected and reported as the produced amount of oil in each period (without the need of any wells).

As mentioned above, depending on the scenario under study, different types of initialization can be considered. The matrix can be initially saturated with any desired amount of oil, water and gas under equilibrium conditions and the pressure of the system should also be specified. The initialization is done by assigning the pressure at the middle layer (which will be the same both in matrix and fracture) and both domains will be initialized using their own hydrostatic head. The matrix will be fully oil-saturated with residual water saturation but if necessary, this is theoretically possible to use any desired initial saturation as well (as long as the system is in equilibrium).

The fracture domain, depending on the desired scenario can have only one of these three states during the whole calculation time: either is fully saturated with water, gas or undersaturated oil and needless to mention that these are for modeling submerging the matrix block in each of thsse mentioned fluids; and the outflow of oil from the matrix domain is extracted from the fracture at the end of each time step, leaving the fracture fluid content the same as it was at the beginning of the time step; this can be done by restoring the fracture content to its original status in terms of fluid content (in moles) and reporting the difference in oil content as the oil production from the matrix block in that time step. Some examples of possible undersaturated scenarios that can be studied using the single matrix block model are mentioned here:

3.4.8.1 Water gravity drainage drive

The matrix domain is saturated with the maximum oil saturation (i.e. $1\text{-}S_{wc}$). The fracture domain is fully water-saturated. There is no capillary pressure defined for the matrix and therefore, the capillary pressure is zero. There will be a hydraulic pressure gradient difference between the fracture domain with water pressure gradient and the matrix domain with oil pressure gradient that drives water into the matrix domain from the bottom, pushing the oil out to the fracture domain from the top. The oil outflow from the matrix to the fracture will be collected as the recovery, leaving the fracture fully water-saturated.

3.4.8.2 Gas gravity drainage drive

The matrix domain is saturated with the maximum oil saturation (i.e. $1\text{-}S_{wc}$). The fracture domain is fully gas-saturated. There is no capillary pressure defined for the matrix and therefore, the capillary pressure is zero. There will be a hydraulic pressure gradient difference between the fracture domain with gas pressure gradient and the matrix domain with oil pressure gradient that pours the oil out from the matrix to the fracture domain from the bottom while gas enters the matrix from the top. The oil outflow from the matrix to the fracture will be collected as the recovery, leaving the fracture fully gas-saturated.

3.4.8.3 Water imbibition drive

The matrix domain is saturated with the maximum oil saturation (i.e. $1\text{-}S_{wc}$). The fracture domain is fully water-saturated. Matrix now has an imbibition capillary pressure function. The water will imbibe from the fracture domain into the matrix domain displacing the oil out of the matrix domain. The oil outflow from the matrix to the fracture will be collected as the recovery, leaving the fracture fully gas-saturated. Note that in this case, water gravity drainage drive is also active and works in favor of extracting oil from the matrix domain; in order to just study the effect of water imbibition, the gravitational drive can be artificially cancelled by setting the same value for the water and oil densities so that the only active driving force would be the capillary imbibition, or to decide to study the effect of both drives simultaneously.

3.4.8.3.1 No phase segregation assumption required

In the conventional model, in order to analytically calculate the gravitational force, an inevitable assumption is made: The phases are completely segregated in both the matrix and the fracture cells as explained in details in Section E.4.2. Use of single matrix block analysis completely removes the need for such assumption as the gravitational effect, is simulated and considered in the final recovery curve. To illustrate this advantage of using single matrix block recovery curve for considering the gravitational force over the analytical method: consider a single matrix block, having a vertical water-filled fracture neighbor as shown in Figure 3.10. In order to calculate the gravitational drive in the conventional model, the block saturation distribution after a certain time is assumed to be as shown in Figure 3.15 which is clearly neither physically nor theoretically correct. The physically correct representation is shown in Figure 3.17 which is actually a display of how the recovery curve is calculated considering the correct saturation distribution. It is noteworthy that the single matrix block model, also considers the complicated interactions of the gravitational force with the capillary force (which is observable in the transition zone in Figure 3.17). This is in no way possible in the conventional approach to be calculated analytically. This is a big theoretical advantage in favor of using the recovery curve to calculate the matrix-fracture transfer (as proposed in this work) rather than the conventional transfer term formula used in the conventional dual-porosity model. The same also applies for gas-filled fractures.

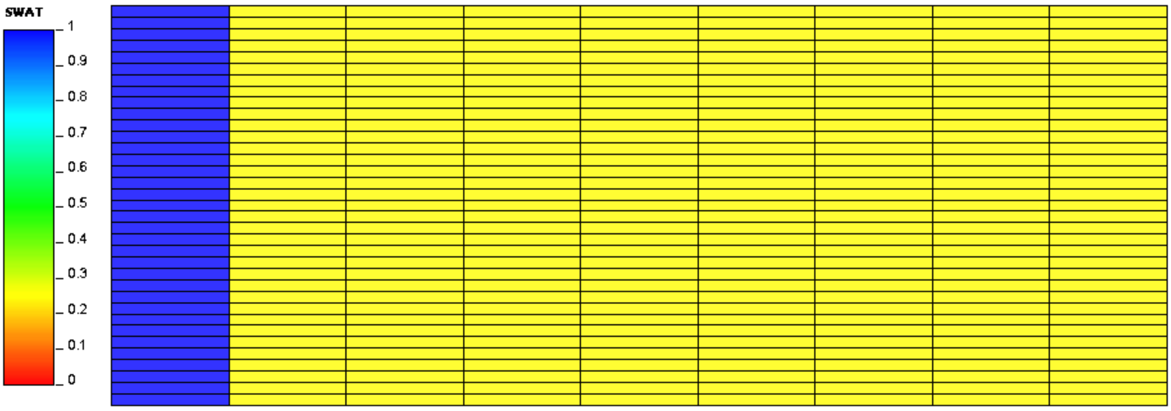


Figure 3.10: Single Matrix Block with a vertical fracture

3.4.8.4 Water viscous displacement drive

The matrix domain is saturated with the maximum oil saturation (i.e. $1\text{-}S_{wc}$). The fracture domain is fully water-saturated. The matrix may or may not have a capillary pressure while the gravity drainage is active in the vertical direction. An additional pressure gradient (due to the viscous displacement) in the fracture domain is imposed to the system in the desired direction(s): the pressure of the mid-fracture cell (in the desired direction) will be the same as the initialization pressure, and the pressure gradient will be added in the positive direction and subtracted in the negative direction from the center cell. This extra pressure difference, drives the water from the fracture domain into the matrix domain in the upstream and the oil out from the matrix domain to the fracture domain in the downstream.

3.4.8.5 Gas viscous displacement drive

Similarly to the previous case, the matrix domain is saturated with the maximum oil saturation (i.e. $1\text{-}S_{wc}$). The fracture is fully gas-saturated. The matrix may or may not have a capillary pressure (which would act against the oil flowing out of the matrix block) while the gravity drainage is active in the vertical direction. An additional pressure gradient (due to the viscous displacement) in the fracture domain is imposed to the system in the desired direction(s): the pressure of the mid-fracture cell (in the desired direction) will be the same as the initialization pressure, and the pressure gradient will be added in the positive direction and subtracted in the negative direction from the center cell. This extra pressure difference, drives the gas from the fracture domain into the matrix domain in the upstream and the oil out from the matrix domain to the fracture domain in the downstream. This can be thought of as submerging the matrix block in flowing water with a velocity that causes the specified pressure gradient across the desired direction(s) in the matrix block.

3.4.8.6 Solution gas drive study

Both matrix and fracture domains are filled with undersaturated oil. The pressure in the fracture is dropped at a specified rate. The amount of oil which is expelled from the matrix domain is collected from the fracture and plotted versus time to represent the transient solution gas drive effects.

To study the solution gas drive under pseudo-steady state, the pressure is dropped at small steps, and the system is allowed (given enough time) to reach the steady state after each pressure drop step. The total amount of oil which leaves the matrix domain in that pressure drop step is plotted versus the pressure drop step number (which is not a recovery curve in this case).

3.4.8.7 Capillary hysteresis and wettability study

Different matrix and fracture initial saturations can be studied with different rock types (water-wet, oil-wet, mixed-wet) with or without capillary hysteresis. The fracture content can be switched from full saturation of one fluid to full saturation of another one (after certain time) to study the effects of capillary hysteresis.

3.4.8.8 Oil resaturation study

The matrix is saturated with maximum gas saturation (or also by maximum water saturation for oil-wet rocks). The fracture is filled with oil and the amount of oil which enters the matrix domain from the fracture domain is reported and plotted as a *Resaturation curve* which is similar to a recovery curve, but instead of the oil extracted from the matrix domain, the oil which enters it is plotted in time.

3.4.9 SMB Model Validation

To demonstrate the applicability of the proposed method, the numerical results of the single matrix block calculations are compared to well-known analytical methods for different production scenarios.[80] The Pirson[82] method serves as reference for solution gas drive, and the Buckley-Leverett[16] solution for water displacement.

3.4.9.1 Solution gas drive - Pirson method

In order to ensure the applicability of the model for reservoirs produced by solution gas drive, numerical results are compared with the analytical solution to the Pirson[82] method. To fulfill the assumptions of the Pirson method in the simulation model, communication in *z*-direction is only allowed between the outer matrix grid cells and the adjacent fracture plane, and along the vertical fractures. All other transmissibilities in *z*-direction are set to zero.The displacement starts at the bubble-point pressure. As the pressure drops, gas is liberated in the matrix and the displaced oil flows toward the fractures. Figure 3.11 shows the results of the analytical and the numerical calculations. The good accordance with the analytical solution proves the applicability of single matrix block models to the investigated expansion-drive scenarios.

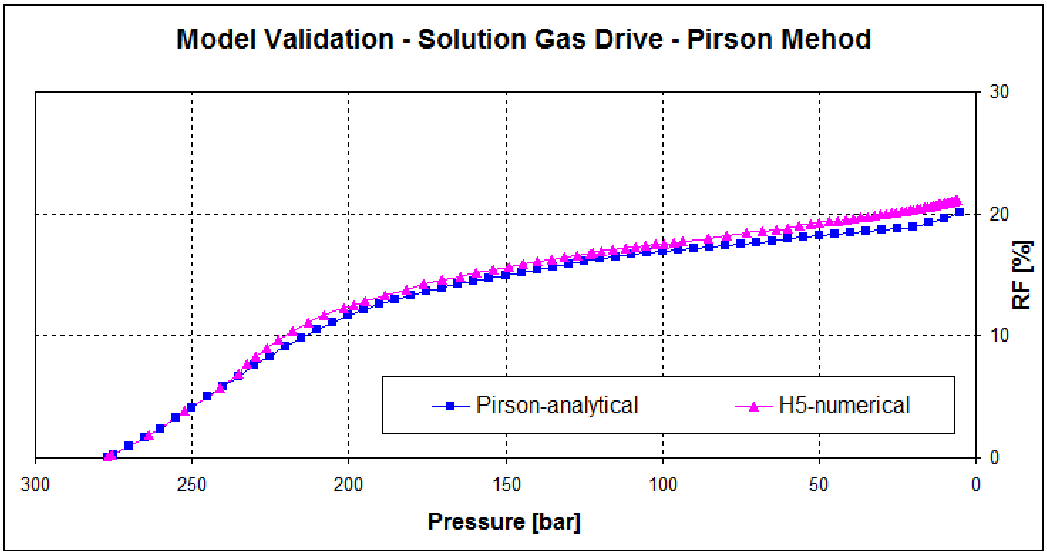


Figure 3.11: Model validation solution gas drive

3.4.9.2 Water displacement - Buckley-Leverett solution

The Buckley-Leverett[16] theory is used to calculate the displacement efficiency in case of water displacement. This theory proves that the displacement can proceed as a front and provides a clear description of how the phase mobilities and gravity influence this displacement. Assumptions in the Buckley-Leverett theory are:

- The two fluids are immiscible and incompressible.
- The porous medium is homogeneous.
- The displacement is one-dimensional and stable.
- The filtration can be described by the multi-phase Darcy's law.
- The capillary force is negligible.

To make the results of the numerical and analytical calculations comparable, these assumptions have to be fulfilled in the simulation model as well. Only the fracture planes at the top and the bottom of the matrix block are set active to ensure one-dimensional frontal displacement. To simulate the frontal displacement, water is added (injected) to the bottom fracture layer and oil is produced from the top layer. The capillary pressure in the matrix is set equal to zero.

Three models with different grid resolutions are calculated to verify the decision of using the *8*x*8*x*36* grid model. Figure 3.12 shows the results of the analytical and the numerical models. For the coarsest grid model, the final recovery from the matrix block matches the analytical solution very well, but the plot of the recovery factor versus the injected pore volume shows some deflection between *0.5* and *1.5* injected pore volumes. Refining the grid by a factor of two gives much better results in this region. An additional refinement results in a slightly better match, but unfortunately the runtime is increased approximately by a factor of *20*.

These results underline the decision of using the *8*x*8*x*36* grid model in terms of optimal accuracy and run time. The results also show that the single matrix block model is capable of representing water displacement in a matrix block.

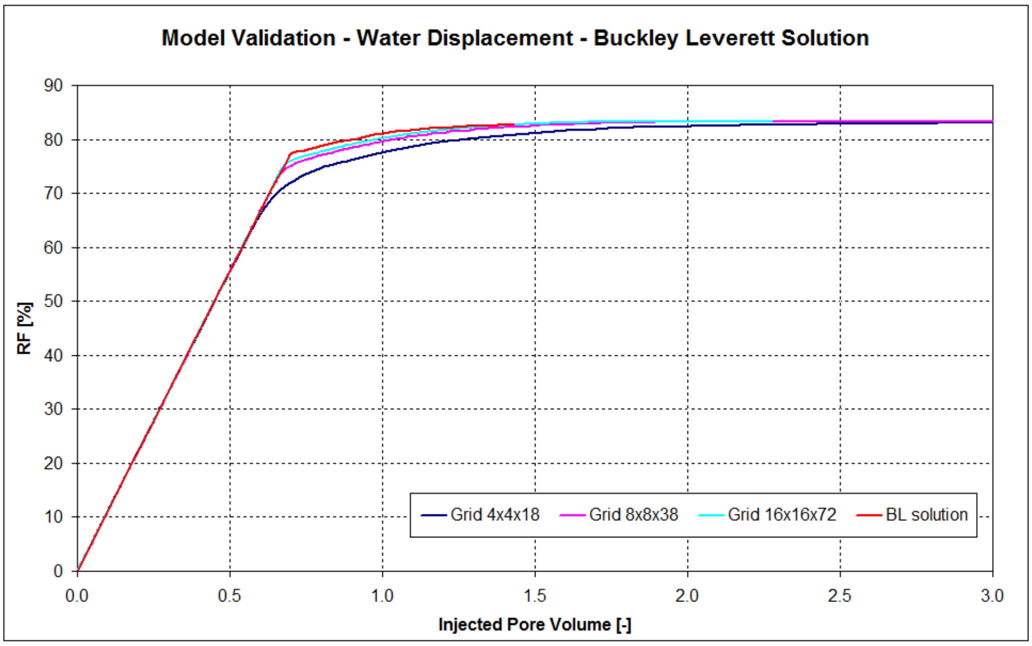


Figure 3.12: Model validation water displacement (from Pirker[81])

3.5 Investigation of the Transfer Term

In order to study the transfer term, a series of single matrix block analyses (see Section 3.4) of an oil-wet matrix block from a full field with the type described in Table 6.2, PVT properties of Table 6.1 and SCAL data of Figure 6.1 and Figure 6.2 is performed under the following five different conditions:

1. fully submerged in gas,
2. fully submerged in gas with only vertical fractures (and no horizontal fractures),
3. fully submerged in water,
4. fully submerged in water with only vertical fractures (and no horizontal fractures),
5. fully submerged in water considering the permeability anisotropy.

Note that in cases 1 to 4, the apparent permeability value (the geometric mean of the directional permeability values) is used in all directions, since the conventional transfer function (as shown in Equation 2.8) assumes only the isotropic permeability ignoring the anisotropy effect; and in the 5th case, this assumption is also put to test.

3.5.1 Results

The recovery curve (i.e. the recovery factor vs. time) from the matrix block to the fracture for each of the five test cases is plotted in Figure 3.13. Since the matrix block is oil-wet, the ultimate recovery from gas drive is much higher than the water drive (due to a larger gravitational force for the gas drive than the water drive).

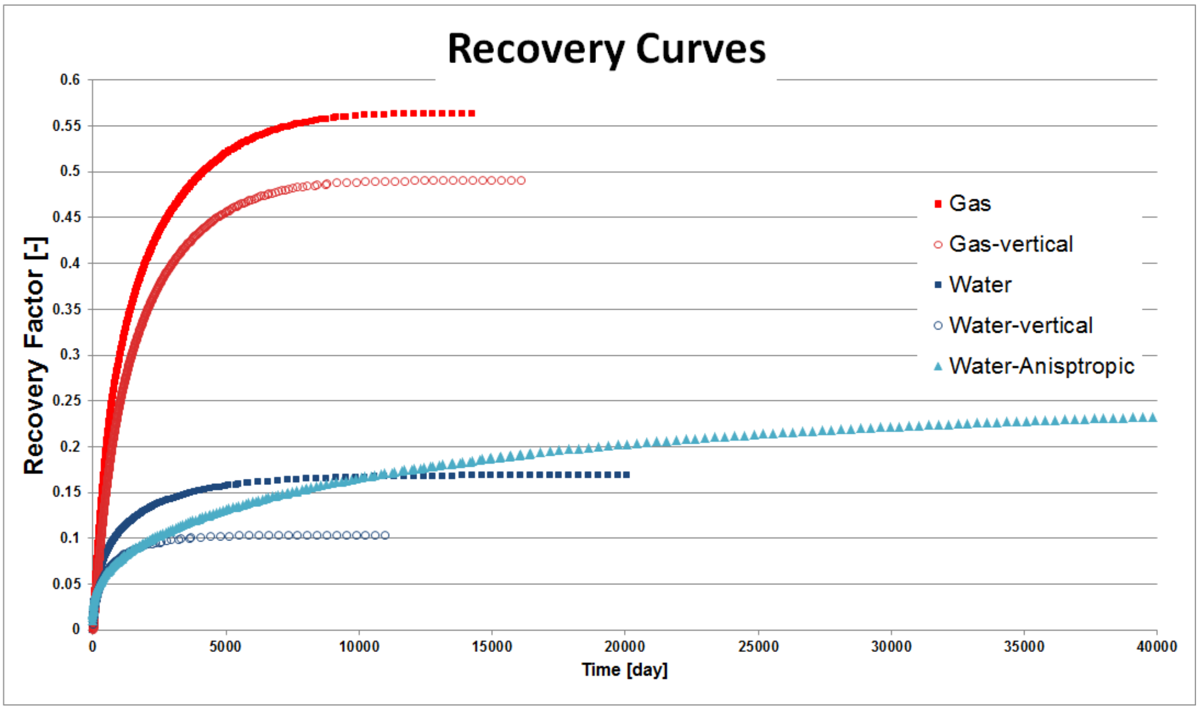


Figure 3.13: Recovery curves for all the test cases

It is noteworthy to consider the significant difference between the third and the fifth case which demonstrates the effect of anisotropy on the matrix-fracture transfer. In this case, the isotropy assumption results in about 10% less oil recovery and a much earlier abandonment time (i.e. the ultimate recovery time) than the actual anisotropic case. This fact, on its own, shows that since the conventional transfer term does not consider the anisotropy, it can lead to completely irrealistic predictions in terms of both the amount and time of recovery from the matrix block.

The results show that ignoring the effects of the horizontal fractures for this example, results in more than 10% less ultimate recovery than when considering the horizontal fractures (for both water and gas cases). On the other hand, in the conventional transfer function, when the gravitational drive is analytically calculated, only the vertical fractures are actually considered and it is assumed that the fracture and matrix blocks are located side-by-side and complete phase segregation occurs immediately in both matrix and fracture (see Figure E.9 in Appendix E). Both of these assumptions are obviously wrong and sources of further discrepancy.

3.5.2 Calculated Shape Factors

Figure 3.14 plots the calculated shape factors from the conventional transfer equation at every time step for each of the five test cases. This can be done by solving Equation 2.8 for σ:

$$\sigma = \frac{q_{cmf}}{V_{cell} k_a \sum_{p=1}^{P} \Lambda_{pc} \bar{\bar{k}}'(\Phi_{pf} - \Phi_{pm})} . \qquad (3.16)$$

As it was earlier mentioned, based on the conventional model assumptions, it is expected to see only one single constant value for the shape factor for all the cases in all times on this plot, since this single value of the shape factor is used in the conventional transfer equation for all the cases. However, the most outstanding observation in Figure 3.14 is that:

> *Contrary to the assumption in the conventional dual-porosity models, the conventionally calculated shape factors are anything but constant!*

In Figure 3.14, the actual constant shape factor value of 0.1 m^{-2} in this example for a certain matrix cell (which was also used in the full-field simulation) is plotted in back dashed line. For all the test cases, at the initial time steps, the calculated shape factors are larger than the actual value at least by a factor of ten and they continuously change in time and go down as low as about four orders of magnitude below the initial value at the point of ultimate recovery (for these cases).

This means that in order to get the same flow rate out of the matrix block in the full field simulation as the fine-scale single matrix block model, the shape factor should change at each time step according to the plots in Figure 3.14 for each one of the cases. In other words, just using a constant value of shape factor for calculation of all boundary conditions, fluid types etc. cannot possibly describe the physical processes that go on between the matrix and the fracture.

However, by a more detailed investigation of the model, it will be shown that the real problem with the discrepancy of the conventional approach is not the shape factor, but actually using the conventional transfer term beyond its limitations. In the next section, it will be shown that the commercially applied methods to calculate the conventional transfer equation under which conditions can be relied upon and under which conditions should not be totally relied on.

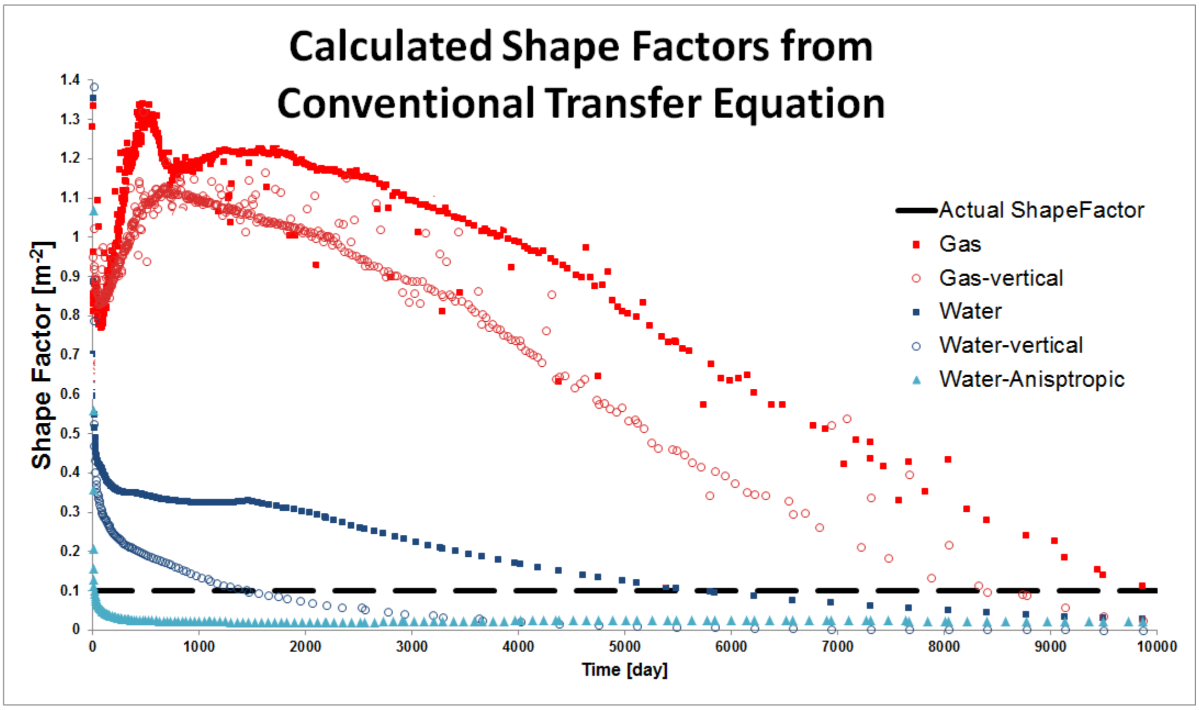


Figure 3.14: Calculated shape factors versus time

3.5.3 Discussion

The conventionally assumed saturation distribution is displayed in Figure 3.15. However, the realistic saturation distribution for water imbibition tests with and without considering the horizontal fractures (test cases 3 & 4), at a certain point in time are shown respectively in Figure 3.16 and Figure 3.17.

It is obvious that the conventional representation (Figure 3.15) is absolutely irrelevant to the reality of the saturation distribution in the matrix block. But what should be even more emphasized on here is that the matrix water saturation near the fracture is much higher than the mean matrix water saturation value. In the conventional approach, the mean matrix saturation is used to calculate the capillary force between the matrix and the fracture, however, in the fine scale model, the capillary force is calculated based on those actual higher water saturations of the matrix cells closer to the fracture (since the water saturation in the matrix cells near the fracture is higher than the average). The same fact also makes analytical calculation of the gravitational force impossible considering the real saturation distribution. This results in completely different driving forces when the average saturations are used comparing to the case when the actual saturation distributions are considered and can be regarded as *the number one cause of the apparent discrepancy in the conventional dual-continuum model*.

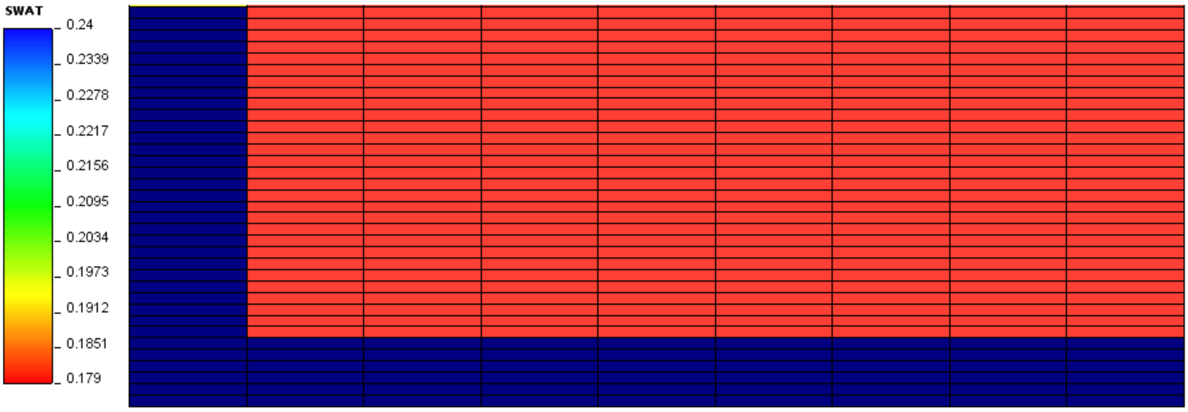


Figure 3.15: Saturation distribution for a certain time in the conventional method

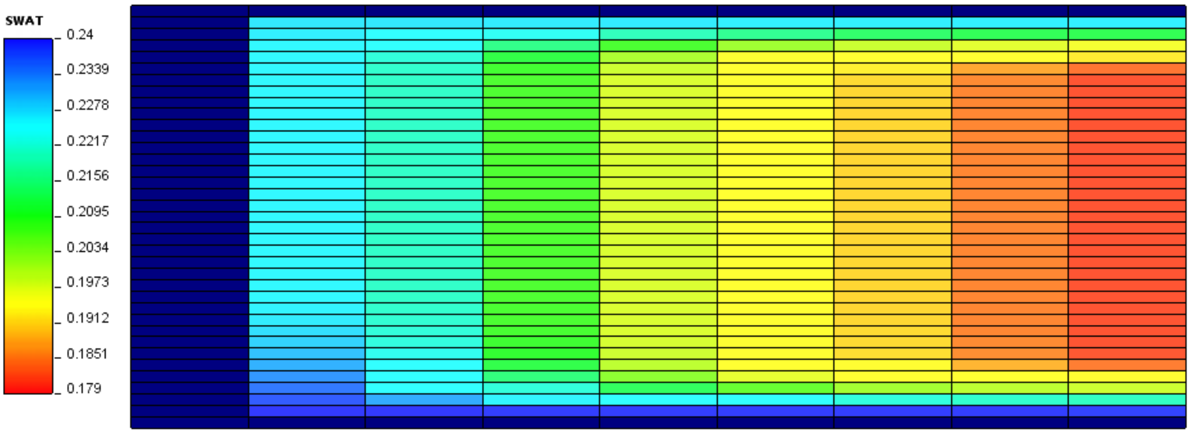


Figure 3.16: Saturation distribution for a certain time for Case 3

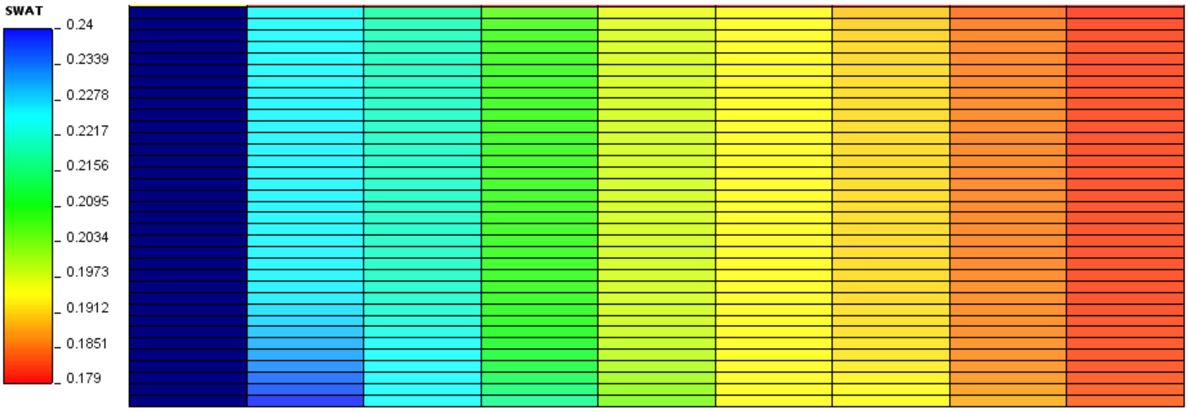


Figure 3.17: Saturation distribution for a certain time for Case 4

On the other hand, in the conventional approach, both the gravitational and imbibition drives are calculated ignoring the effects of horizontal fractures: it is not possible to analytically calculate the simultaneous co-current and countercurrent imbibition along with the sidewise imbibition which occurs when considering the horizontal fractures neither is it possible to analytically calculate the gravitational force for such boundary conditions.

Therefore, as explained in Appendix E, when calculating the potential difference between the matrix and the fracture, the horizontal fracture planes are ignored in the conventional approach. This is another source of discrepancy in the conventional approach: the outstanding difference in saturation distribution pattern as well as the average saturation values when considering or ignoring the horizontal fractures can be seen in Figure 3.16 and Figure 3.17. It should be noted that no matter how different the saturation distributions between these two cases are, the conventional approach calculates the potential difference for both of them using in exactly the same manner. The extent to how irrealistic such assumptions are can be seen in Figure 3.14 by how differently the shape factors are calculated at any time step for cases 1 & 2 and 3 & 4.

The conventional transfer term, completely ignores the anisotropy effect (and simply uses one absolute permeability value in the transfer equation) which is in turn another important source of discrepancy which can be observed on Figure 3.13 (cases 3 & 5) on the recovery pattern (and even the ultimate recovery) and in Figure 3.14 (cases 3 & 5) on how differently the calculated shape factors would be if anisotropy is considered or ignored. The effect of anisotropy on the saturation distribution for cases 3 and 5 can be observed in Figure 3.18 and Figure 3.19 respectively.

What can be observed in Figure 3.15, Figure 3.16, Figure 3.17, Figure 3.18 and Figure 3.19 is that the critical deficiency in the calculation of conventional transfer equation is inadequacy of the analytical approaches to calculate the matrix-fracture potential difference and not the shape factor! The analytical methods for calculation of the potential difference depend upon the assumption of segregated fluid saturation distribution in the matrix block which. Although this assumption is practically acceptable for calculating the fluid properties (e.g. oil viscosity) but in this work it was shown that using the average saturations for calculating the potential difference can bring in very large sources of discrepancy which can in no way be blamed on the definition of the shape factor!

This study shows that:

> *the conventionally practiced method is not reliable to calculate the potential difference in case of displacement of one fluid by another one (when the saturation distribution would become inhomogeneous), it also shows that in cases where the saturation distribution is homogeneous (e.g. expansion drive or single phase transfer), the conventional transfer equation can be reliably used.*

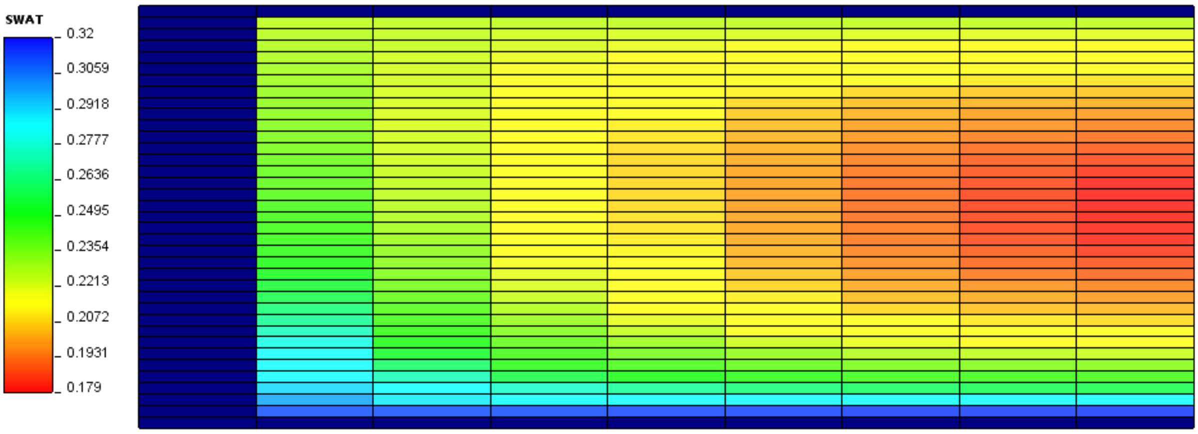


Figure 3.18: Saturation distribution for a certain time for Case 3 (isotropic)

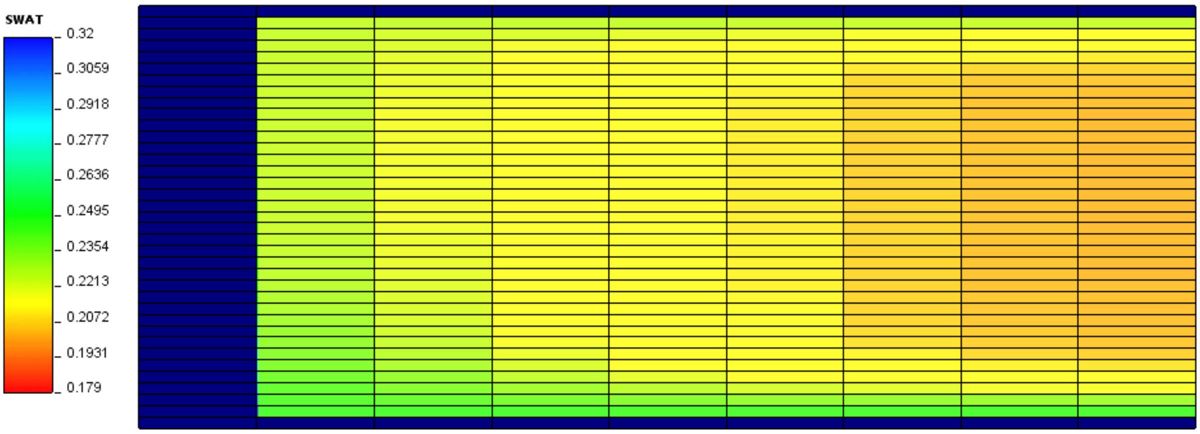


Figure 3.19: Saturation distribution for a certain time for Case 5 (anisotropic)

3.6 Matrix Block Identification Methods

The matrix block shapes can be identified from any method which is available and that can be used. In this section, two methods are suggested.

3.6.1 Discrete Fracture Models

The matrix block shapes can also be extracted from the discrete fracture models. The corner points can be used to build the 3D model of all matrix blocks which are modelled in the reservoir. This data can be scanned to find the occurrence of similar matrix block IDs at different locations in the reservoir to produce the *matrix block ID distribution* in the reservoir.

Normally when speaking of the DFM, the image that comes to the mind, is like what is shown in Figure 2.3, which is actually a correct one (as it displays the individual fractures that are modelled in the DFM). However, what is less focused on in this image, are the matrix blocks existing in between the fracture planes or in other words: the complement of the fracture planes.

Figure 3.20 is an illustration of how matrix blocks would look like in a DFM, if instead of the fractures, the matrix blocks were to be drawn. It clearly shows how it is possible to use DFM to know the matrix block shapes and their distribution in the reservoir (and in the upscaled simulation cells).

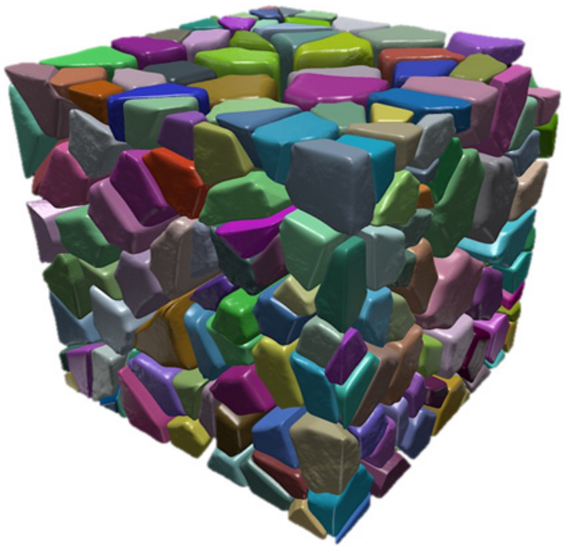


Figure 3.20: Representation of matrix blocks extractable from Discrete Fracture Models (by courtesy of yuku.com[50])

3.6.2 Outcrop Investigation

Outcrop investigation can be used to obtain the shape of the matrix blocks as shown in Figure 2.6 and Figure 3.21 for the formation. The information gathered from the outcrop study about the surface matrix blocks and fracture sets, can be used in the geological models to estimate the fracture network (and the matrix block shapes) in the reservoir statistically.

Figure 3.21: Outcrop of a fractured formation (from Lash[55])

3.7 Cell Recovery Curve from Conventional Simulation

It is possible to monitor a cell in the conventional dual-porosity simulation for changes of the (molar or volumetric) fluid contents and calculate the amount of recovered oil from it versus time (i.e. the recovery curve) using Equation 3.17:

$$RF(t) = \frac{S_o(t) \cdot B_{oi}}{S_{oi} \cdot B_i(t)} - 1. \tag{3.17}$$

This feature is implemented as a standard feature in different commercial reservoir simulators e.g.[45],[95] and it is possible to specify the matrix cells in the conventional simulation run, to output the recovery curve versus time as well as the additional information about the pressure and saturation changes in the matrix cell and the adjacent fracture cell. In Section 6.3 an example of using this method to produce the recovery curve from a conventional simulation and then using it for the recovery curve simulation is demonstrated.

3.8 Calculation of the Matrix-Fracture Transfer Rate from a Recovery Curve

In order to calculate the transfer term from the recovery curve in each time step, the increment in recovery is read from the recovery curve based on the current recovery factor and the time step length as will be explained in this section.

The original oil in place in moles of the matrix cell *I* is calculated volumetrically:

$$OOIP_I = V_{\phi i} \cdot S_{oi} / B_{oi}, \tag{3.18}$$

where $V_{\phi i}$ is initial the pore volume of the block, S_{oi} is the initial oil saturation in the block and B_{oi} is the initial oil formation factor. The actual oil in place, OIP_I, in moles contained at the beginning of the time step, *t*, in the matrix block *I* is:

$$OIP_{I,t} = V_\phi \cdot (1 - S_w - S_g) / B_o. \tag{3.19}$$

The pore volume V_ϕ is the pore volume and ($1\text{-}S_w\text{-}S_g$) is the oil saturation of block *I* at the beginning of the time step *t*. The recovery factor of the block at the beginning of each time step is:

$$RF_I(t) = 1 - \frac{OIP_{I,t}}{OOIP_I}. \tag{3.20}$$

The increment in the recovery from the matrix block is calculated as illustrated in Figure 3.22, starting from the known recovery factor, $RF_I(t)$, and finding the new recovery factor, $RF_I(t+\Delta t)$, by adding the time step length Δt to the *virtual time* axis on the recovery curve. This is called the virtual time because the time value on the recovery curve horizontal axis corresponding to the $RF_I(t)$ is not the same as the actual simulation time, t. But only the recovery curve increment for the time step on the curve is interesting, and not the absolute time.

$$q_{mf}^{block^I}(t, \Delta t) = OOIP_I \times (RF_I(t + \Delta t) - RF_I(t)) / \Delta t\ . \tag{3.21}$$

The volumetric matrix-fracture transfer rate for the time step is then calculated from Equation 3.21 which can readily be used instead of Equation 2.8 in the conventional dual-porosity formulation (Equation 2.6 and Equation 2.7). In other words, in this approach nothing in the conventional model needs to be changed, except the matrix-fracture transfer term.

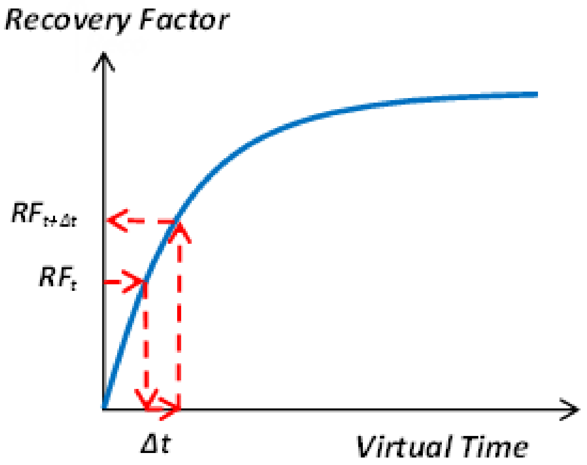


Figure 3.22: Estimation of the time and recovery factor at the beginning of the time step.

3.9 Lumping Recovery Curves

As described in the previous section (Section 3.8), the recovery curves represent the oil outflow *rate* from the matrix block (under a certain set of conditions) from a known recovery factor (state) parameter at any time for a known interval of time. In other words, Equation 3.21 can be rewritten in the volumetric form as shown in Equation 3.22 where $V_o^I(t, \Delta t)$ is the volume of oil that was extracted from the matrix block *I* in the time interval $(t, t + \Delta t)$ in standard cubic meters:

$$V_o^I(t, \Delta t) = q_{mf}^{block^I}(t, \Delta t) \times \Delta t \qquad . \tag{3.22}$$

$$V_o^I(t, \Delta t) = OOIP_I \times (RF_I(t + \Delta t) - RF_I(t))$$

If there are different matrix blocks under the same condition, the amount of oil which is extracted from each one can be calculated from Equation 3.22 based on its own recovery curve. In a simulation cell, there are several matrix blocks (see Figure 3.20). It is obvious that if the cell conditions can be assumed to be independent of the matrix blocks' productions (at least for the time interval under consideration), the produced moles from the matrix blocks can be summed up to form the total amount of oil extracted from the matrix simulation cell by superposition. In other words, in Figure 3.20, considering (all) the matrix blocks to represent the matrix cell (and the empty spaces in between them, to represent the fracture cell), the amount of oil transferred from the matrix cell to the fracture cell in a time interval, is the summation of all the oil amounts extracted from each individual matrix blocks (in the matrix cell) in that time interval.

This means that the recovery curve of the simulation cell, is simply the weighted average of the recovery curves of the matrix blocks in that cell. It should be "weighted" average because the same matrix block, *can* be several times present in the cell. Knowing the frequencies of each *matrix block*, the cell recovery curve can be calculated from Equation 3.23 and also displayed in Figure 3.23.

$$RF^{cell}(t) = \sum_{blocks} RF^I(t) \cdot f_I \, . \tag{3.23}$$

Lumping recovery curves is not only used to calculate the cell recovery curve from the block recovery curves, but it can be used in any instance which the recovery from some curves are to be collected (and counted) as one entity; for instance, to calculate the *class recovery curves* and *region recovery curves* which will be introduced in Chapter 4. Figure 3.23 shows how Equation 3.23 can be applied on the matrix block recovery curves to create the lumped (average-weighted) recovery curve.

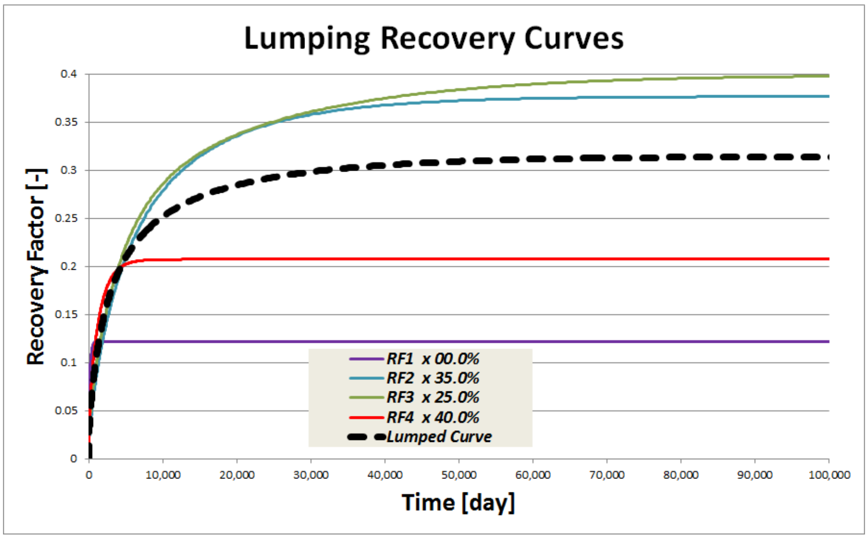


Figure 3.23: Graphical representation of lumping recovery curves

3.10 Dimensionless Time

The component rate given by the Equation 2.8 is a function of the time. The phase potential and the relative permeabilities are function of the saturation, which naturally changes if the water or gas in the surrounding fractures invades the matrix block. Two immiscible and low compressible phases are considered. One of them is the displacing phase the other the displaced one. The composition of the displaced phase can be simplified to a single pseudo-component named as oil, therefore $x_{pc} = x_{oo}= 1$. The cumulative production of the component oil is:

$$Q_{mf}(t) = \int_0^t q_{mf}(t)dt = \int_0^t \frac{k_{ro}D_o}{\mu_o}(\Phi_{of} - \Phi_{om})\sigma k_a dt \quad . \tag{3.24}$$

Divided with the original number of moles within the matrix block, the fracture matrix mass transfer can be expressed also with a time-dependent *recovery factor, RF*:

$$\begin{aligned} RF(t) &= \frac{1}{N_o}\int_0^t \frac{k_{ro}D_o}{\mu_o}(\Phi_{of} - \Phi_{om})\sigma k_a dt \quad . \\ &= \frac{(\phi\mu_o c_o)_{ini}}{N_o}\int_0^t \frac{k_{ro}D_o}{\mu_o}(\Phi_{of} - \Phi_{om})\frac{\sigma k_a}{(\phi\mu_o c_o)_{ini}}dt \end{aligned} \tag{3.25}$$

In Equation 3.25, the constant factor before *dt* has the dimension of one over time. Multiplying this factor with the time value, a dimensionless variable of time is achieved called the *dimensionless time*, t_D shown in Equation 3.26:

$$t_D = \frac{\sigma k_a}{(\phi \mu_o c_o)_{ini}} t = \beta t. \tag{3.26}$$

The recovery function (matrix block recovery factor vs. time) can be written in terms of this dimensionless time to give the *scaled recovery factor,* $\overline{RF}$ assuming that the functions under integral, i.e. $(\Phi_{of} - \Phi_{om})$, are independent of the factors scaling the time variable to the dimensionless time. Note that this assumption holds within it that the driving forces do not change with the factors defining the dimensionless time conversion factor (e.g. the shape factor). Rewriting Equation 3.25 in terms of the dimensionless time yields:

$$\overline{RF}(t_D) = \frac{(\phi \mu_o c_o)_{ini}}{N_o} \int_0^{t_D} \frac{k_{ro} D_o}{\mu_o} (\Phi_{of} - \Phi_{om}) dt_D, \tag{3.27}$$

$$\overline{RF}(t_D) = RF(t_D / \beta), \tag{3.28}$$

where,

t_D is dimensionless time,
k_a is the apparent permeability of the matrix block [m^2],
σ is the shape factor [m^{-2}],
$c_{o_{ini}}$ is the initial, oil compressibility [bar^{-1}]
$\mu_{o_{ini}}$ is the initial oil viscosity [cp] multiplication and,
ϕ is the porosity [-],
β is the dimensionless time conversion factor,
N_o is the initial oil in place [sm^3],
k_{ro} is the oil relative permeability,
D_o is the oil specific molar density [kmol/ sm^3],
Φ_{of} is the fracture oil potential [bar],
Φ_{om} is the matrix oil potential [bar].

Chapter 4

Theory of the Work

4.1 Introduction

In this chapter, the new concept for modeling and simulating naturally fractured reservoirs suggested by this research is presented and explained in details.

4.2 Modeling Simulation Cell

After spatial discretization of the reservoir, the simulation cell (shown in Figure 4.1) is modelled (most-commonly in industrial simulators such as ECLIPSE[95]) with the Warren and Root[99] approach, as illustrated in Figure 4.2.

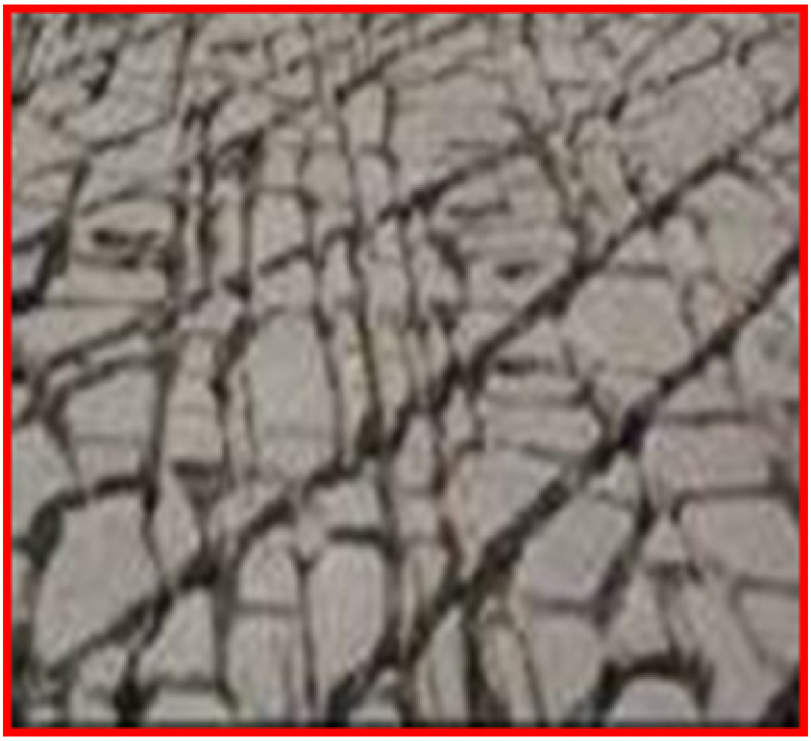


Figure 4.1: Representation of the actual discretized reservoir simulation cell

Comparing Figure 4.2 with Figure 4.1 clearly demonstrates that such homogenization of the matrix *blocks* inside the simulation *cell*, suggested by Warren and Root[99] sugar cube model, is clearly not the best ever representation; and one cannot assign a single value (for all properties, including the shape factor) to all of the matrix blocks in the simulation cell as suggested by this model.

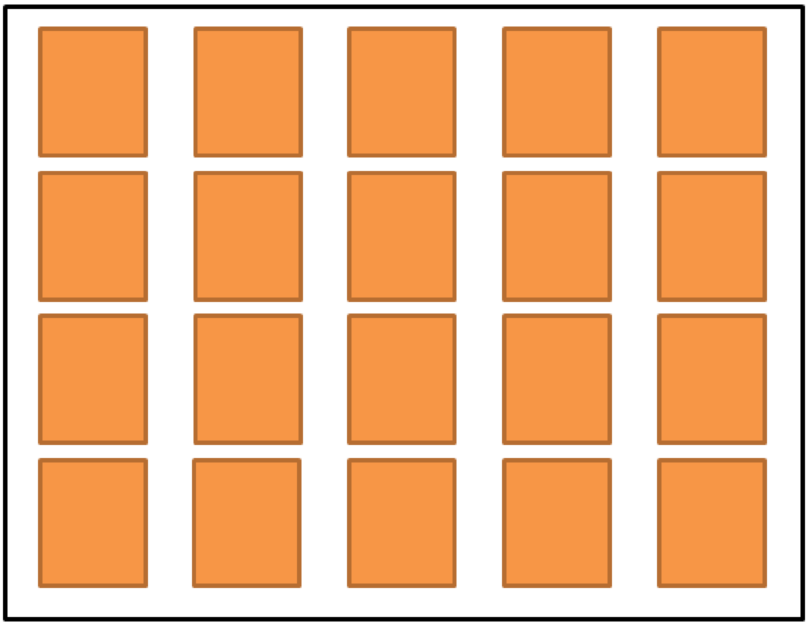


Figure 4.2: Representation of simulation cell in Warren and Root[99] concept

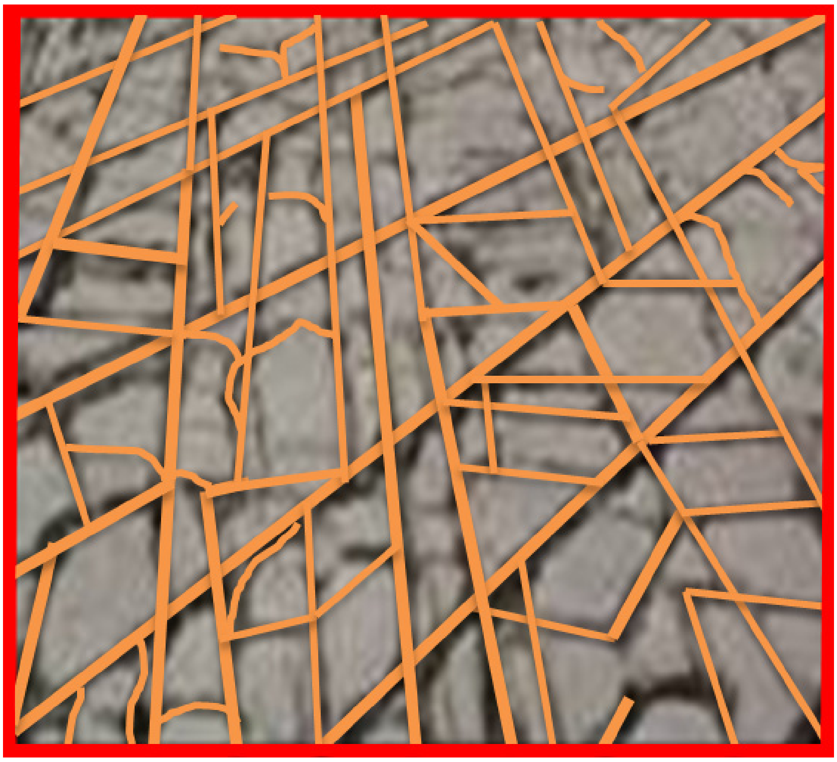


Figure 4.3: Representation of the ideal model

This irrealistic level of idealization and homogenization has ever been known to be physically incorrect, and other methods (such as DFM) have been presented to tackle this problem. But unfortunately, there always have been problems in using other methods on large industrial scales. As can be easily imagined, the best approach *would* be to model the matrix and fractures, *exactly* as they exist in the reservoir (Figure 4.3). But it is a dream too good to become true, since we can never *know* how the fractures really exist inside the reservoir with this level of precision (hundreds of meters under the ground) and even if so, we are far away from being able to economically calculate a model with that level of precision. On the other hand, it is still possible to *statistically estimate* the fracture distribution and create a discrete fracture model (DFM) from fracture network orientation, aperture, size etc. Although DFM cannot have the accuracy represented in Figure 4.3 but can offer a much closer matrix block ID distribution to reality than the sugar cube model (of Figure 4.2).

The advantage of using DFM over the sugar cube model is that instead of homogenizing all the matrix blocks in a simulation cell to the same ideal (cubic) shape, each matrix block shape is calculated statistically from the available input from the actual field and the matrix-fracture transfer could be offered to be calculated in a micro-scale single-porosity boundary-condition approach. As mentioned, this leads to a bigger problem that has prevented the DFM approach to be applied in industry: the fact that this method is too CPU-intensive to be applicable for large reservoirs due to high level of statistically generated details and unknowns.

Pirker[81] followed the suggestion of Heinemann[44] that it is possible and advantageous to use recovery curves to calculate the matrix-fracture transfer term, and proved its applicability on a column model with only one recovery curve for the simulation of the water imbibition of the cells containing matrix blocks of different shape factors. Figure 4.4 is a schematic of the model that Pirker[81] used for the simulation cells. It is assumed that the simulation cell consists of matrix blocks with different shape factors.

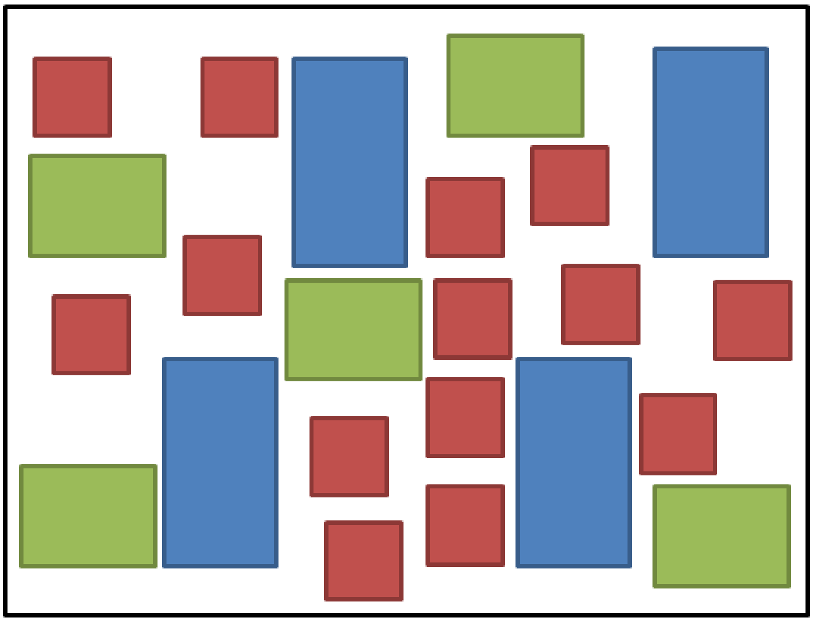


Figure 4.4: Representation of the Pirker[81] model of simulation cell

In brief, what was done by Pirker[81] is explained here:

> In a column model, the recovery from different matrix blocks (with different shape factors) under water imbibition drive was calculated (before the actual run) and then lumped based on the shape factor distribution in the simulation cell; then the recovery of the simulation cell in the actual run would be calculated from this curve. This method was proven to be not only much more accurate than the conventional sugar cube model, but unlike DFM, was shown to be applicable for using in today's industrial NFR simulators (without additional hardware requirements).

In the present work, the Single Matrix Block Analysis (to study the depletion process on a matrix block level) is extended for using in all cases easily with current reservoir simulation models and applied in ***PRS***[45]. The proposed single matrix block model and its usage are explained in detail in Section 3.4. Additionally, while trying to implement the recovery curve approach to calculate the matrix-fracture transfer term that was suggested by Pirker[81], in this work it was observed that although the suggested model (illustrated in Figure 4.4) is much better than the conventional sugar cube model (Figure 4.2), it still needs too much of (pre-processing) data to be really practical in a full field reservoir (and just would remain as a theoretical model too difficult to be practicable). Therefore, a new concept to model the simulation cell was devised which completes the recovery curve approach to simulation of naturally fractured reservoirs and is easily applicable in all mature conventional simulators with minor modification of the equations and practically no additional simulation time.

This concept is illustrated in Figure 4.5: the matrix blocks with the same *scaled recovery curve* have the same color, and all the blocks of the same *matrix block ID* share the same texture (and color). The detailed definition of these concepts are explained in Section 4.3.

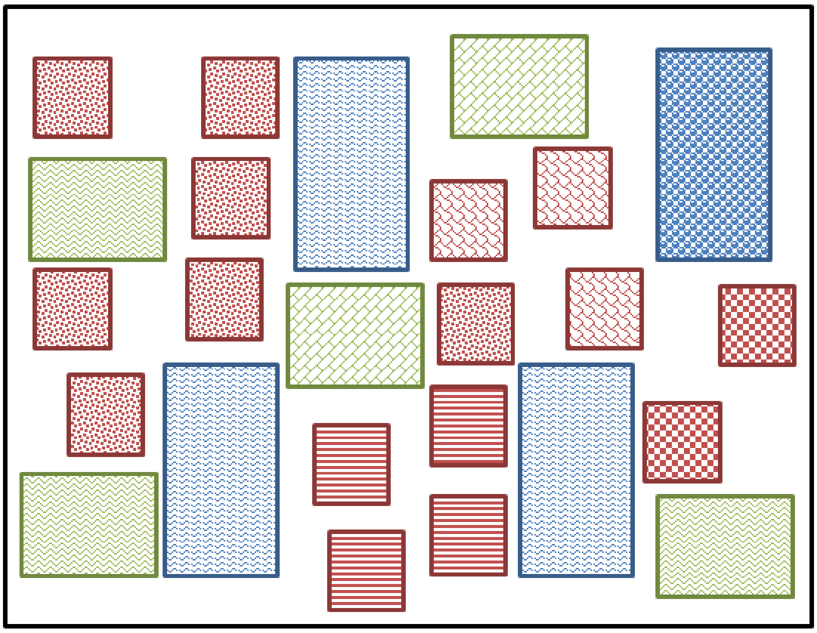


Figure 4.5: Simulation Cell belonging to a Recovery Curve Region

In the new presented concept, the approach is from small-scale to large-scale. In other words, the steps to perform a full field naturally fractured reservoir simulation using recovery curve approach are:

1. Generating the recovery curve (*Recovery Factor* vs. time) for each *matrix block ID* (Section 4.3.1).
2. Using the *dimensionless time* (Section 3.10) to scale the recovery curves (Section 4.3.2).
3. Setting up the *matrix block classes* and determining *matrix block ID distribution* in each class (Section 4.3.3).
4. Defining *recovery curve regions* and assigning one such region to each simulation cell in the model (Section 4.3.4).
5. Simulation of the naturally fractured reservoir with the new method (Chapter 5).

4.3 The new approach to modeling Matrix-Fracture Interactions using recovery curves

In this section, the new proposed model to simulate naturally fractured reservoirs using recovery curves is explained in details. This method is also successfully applied in ***PRS***[45].

4.3.1 *Matrix Block ID* Recovery Curves

In naturally fractured reservoirs, as a result of interaction of different fracture sets, a pattern of (approximately) the same matrix block shape being repeated over a large scale might be observed. In Figure 4.6 two different matrix block shapes occurring in a cell are displayed (in white and yellow colors). Such a shape for matrix blocks is called *Matrix Block ID* if the static properties (such as rock and fluid types, porosity, permeability, etc.) of the blocks are also the same.

Figure 4.4 shows a sugar cube representation of a simulation cell containing different matrix block IDs (shown by different colors) which is a more realistic representation than the original Warren and Root[99] model shown in Figure 4.2.

It is obvious that these blocks will have the same (or very similar) shape factors since they have the same static properties and similar shapes. If the difference of block sizes (or shapes) is large, then they cannot be counted as the same matrix block ID even if the shape factor is the same. The criteria for a matrix block ID, is the following:

> *"The recovery curve under a certain set of conditions is the same for all matrix blocks of the same ID".*

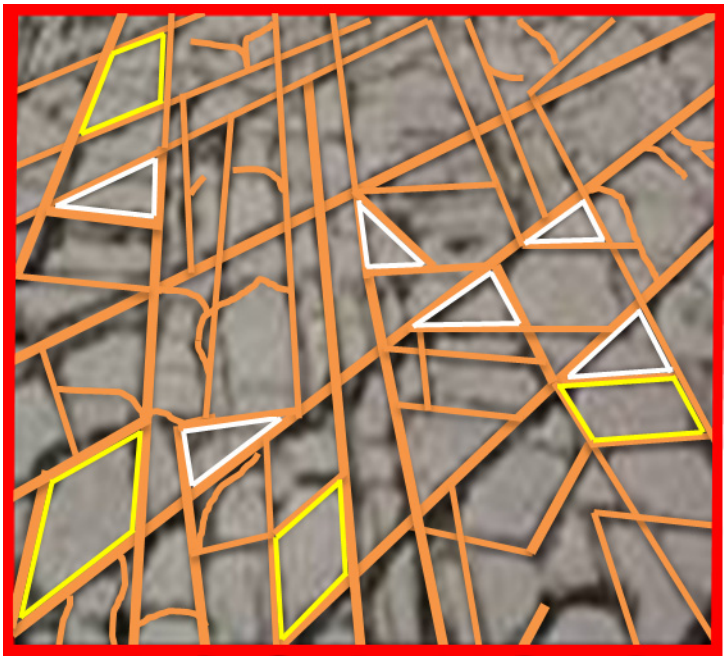


Figure 4.6: Matrix Block IDs

4.3.2 Scaled Recovery Curve

Scaled recovery curve is the recovery curve, scaled on the time axis by the dimensionless time conversion factor, β from Equation 3.26. Figure 4.7 is the plot for three matrix block IDs, with various apparent permeability and porosity combinations while everything else is the same. All these matrix blocks are initially saturated with maximum oil amount and undergo water imbibition and gravity drainage surrounded by fully water-saturated fractures.

However if the same curves are plotted versus the dimensionless time rather than time, all the curves will become *one scaled curve* as shown in Figure 4.8.

The same conditions is repeated for one matrix block ID with three different oil viscosities undergoing gas drainage, as shown in Figure 4.9. However, if the recovery factors are plotted versus the dimensionless time, they would become one curve as shown in Figure 4.10.

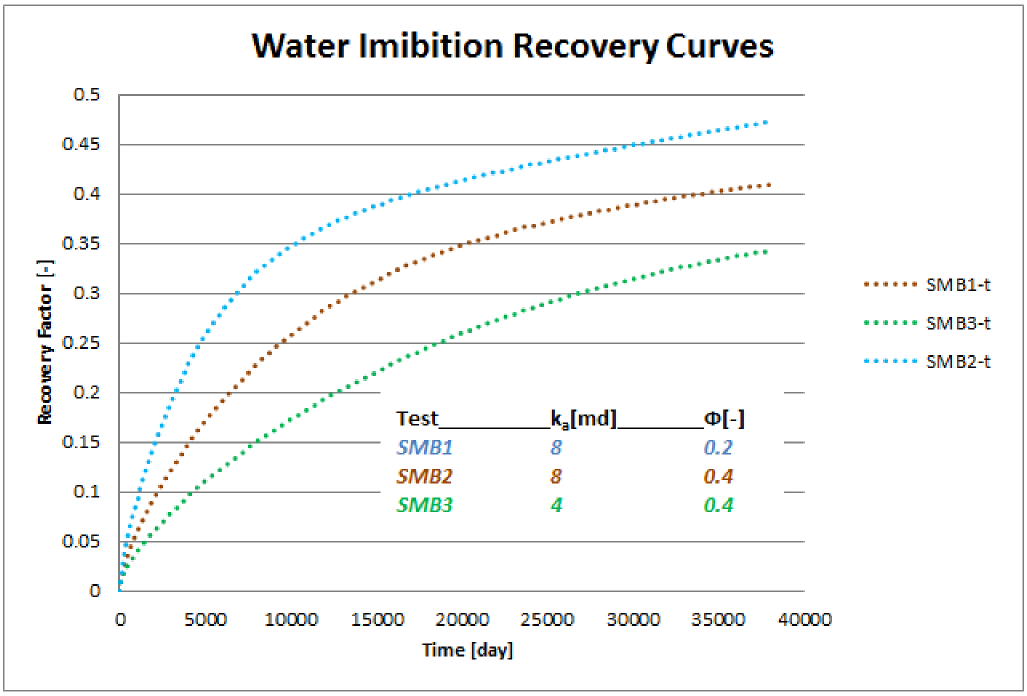


Figure 4.7: Water imbibition recovery curves with different matrix properties

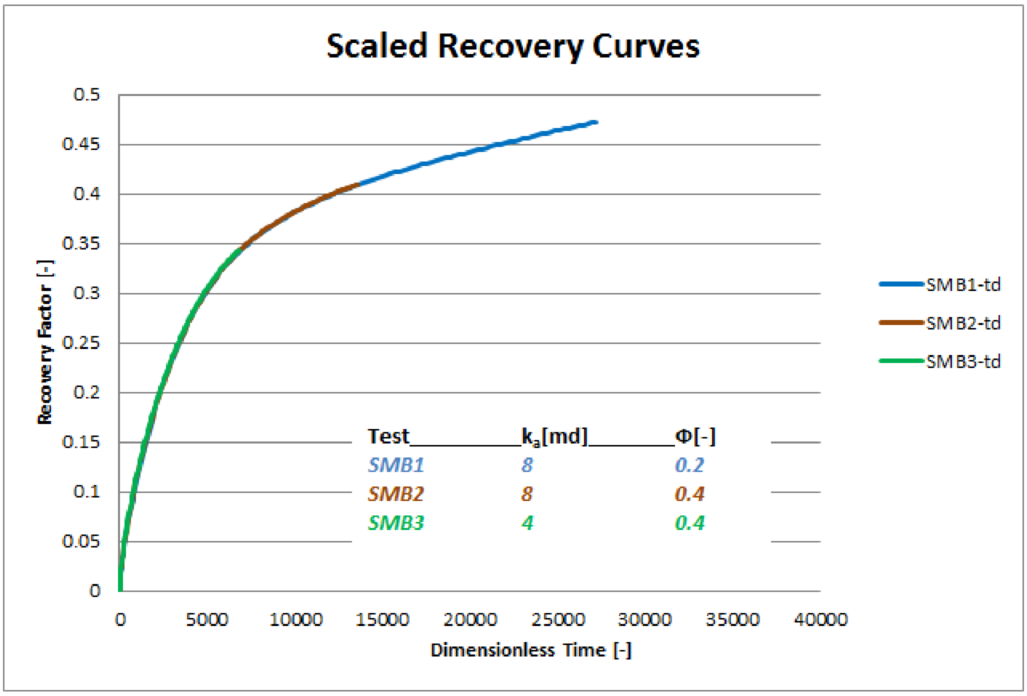


Figure 4.8: Scaled recovery curves for the results in Figure 4.7

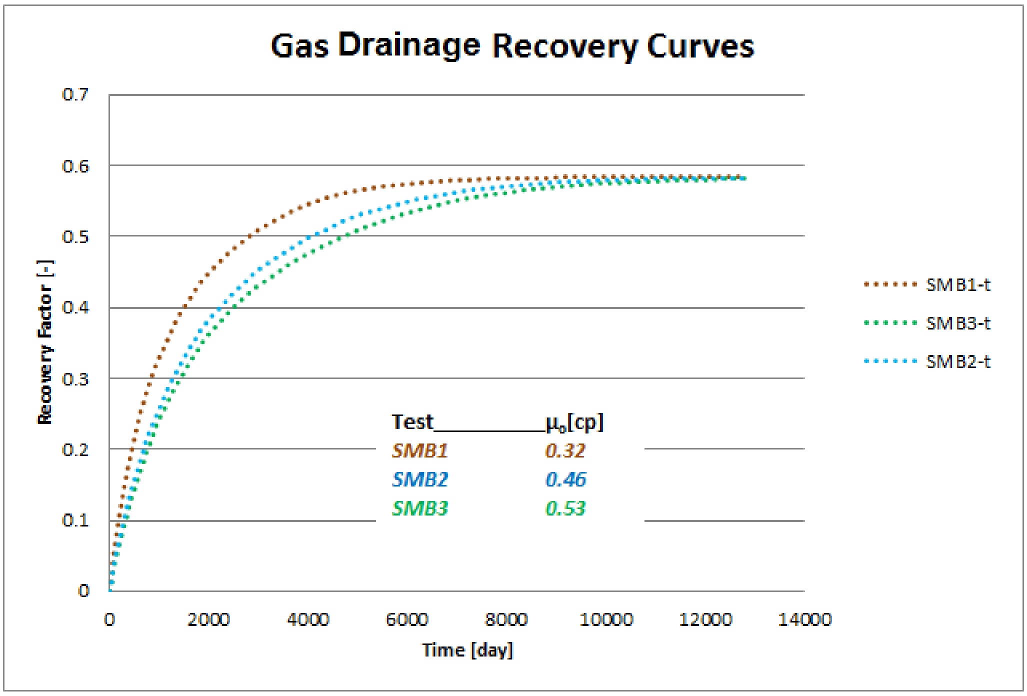


Figure 4.9: Gravitational gas drainage with different oil viscosities

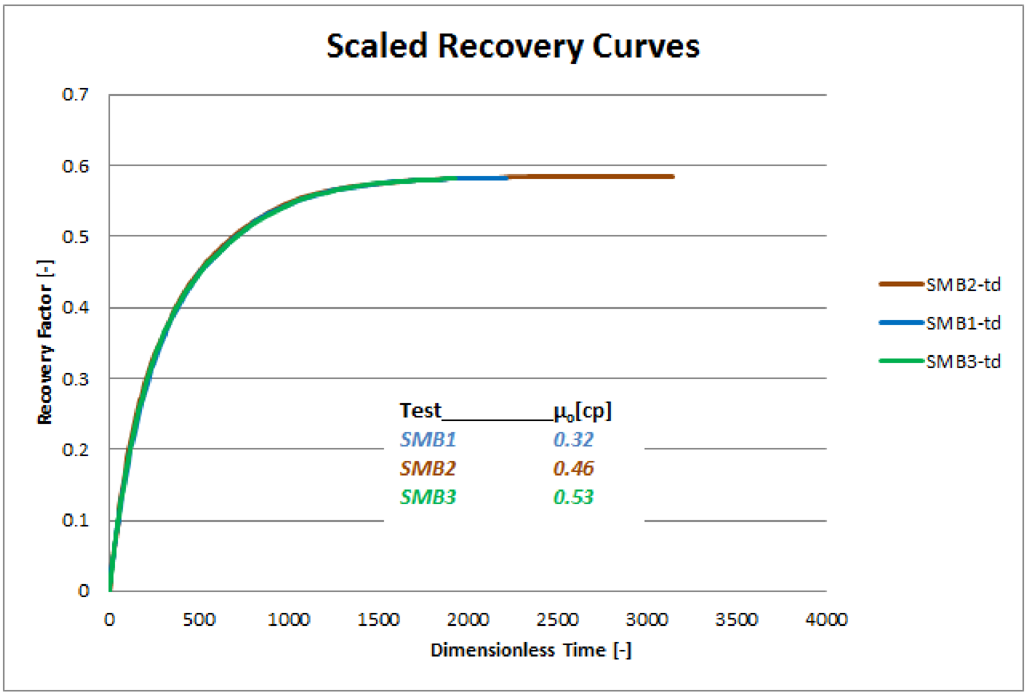


Figure 4.10: Scaled recovery curve for the result in Figure 4.9

4.3.3 Matrix Block Classes

As it was explained in Section 4.3.2, it is possible to represent the production trend of many different matrix block IDs with a single scaled recovery curve using the dimensionless time conversion factor. A *Matrix Block Class* consists of all matrix block IDs of different shapes (i.e. shape factors, σ), porosities, apparent permeabilities, with different oil viscosities and compressibilities that are combined into the same scaled recovery curve, given that the driving forces are the same for all. This means that it is theoretically possible to measure/calculate the recovery curve for one matrix block ID (under a certain set of driving forces) and then scale it using the dimensionless time conversion factor, and to use it for other matrix block IDs in the same matrix block class, which may even have quite different shapes, sizes and apparent permeabilities. So the criteria for creating the matrix block classes would be the following:

> *"The scaled recovery curve under the same driving forces are the same for all matrix block IDs of the same matrix block class".*

4.3.4 Recovery Curve Regions

Since it would be impossible (or impractical) to calculate the recovery curve for each matrix block ID that exists in a reservoir, matrix block classing gives the advantage to assign a *distribution of matrix block classes* to different regions of the reservoir called, *recovery curve regions*.

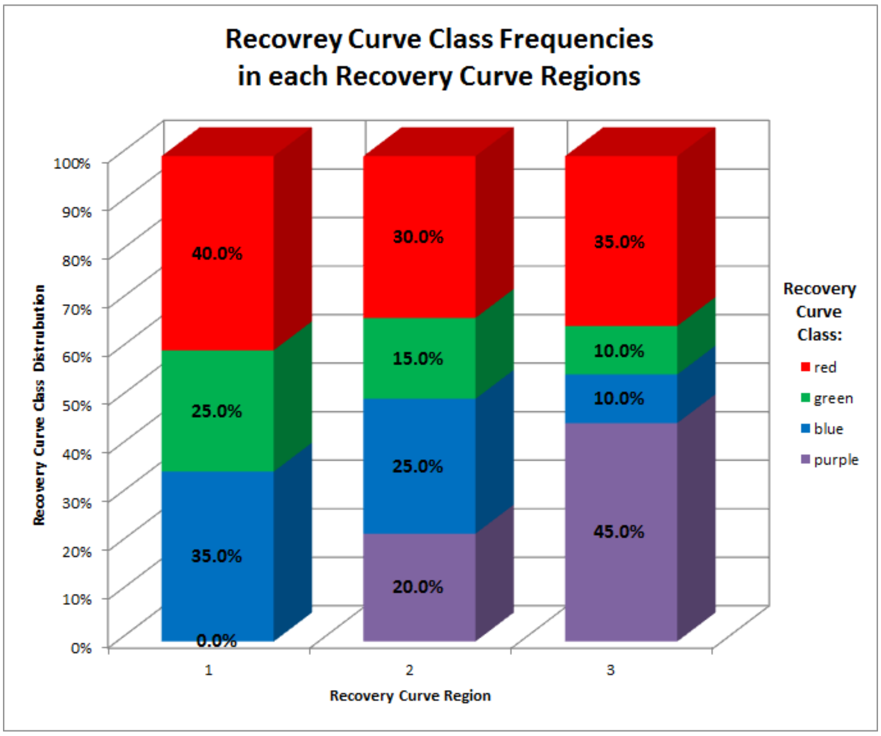


Figure 4.11: Recovery Curve Regions

Figure 4.11 shows an example of a reservoir with three recovery curve regions, each with a certain distribution of the four matrix block classes (of red, green, blue and purple) that exist in the reservoir. Each simulation cell, will be assigned a recovery curve region number (of 1, 2 or 3). Figure 4.5 illustrates a simulation cell, belonging to recovery curve region 1 in Figure 4.11 (which does not contain any matrix blocks of the purple matrix block class).

In Section 4.3.5 it is described in detail how recovery curve regions are set up and how to calculate the simulation cell recovery for each recovery curve region during the simulation run.

4.3.5 Simulation Cell Model

The proposed model for simulation cell is illustrated in Figure 4.5. All the matrix blocks are displayed as colored textured rectangles; however, they do not need to be necessarily rectangles and they can be of any arbitrary matrix block ID as long as the recovery curve for that matrix block ID is known from any possible source (see Section 5.2.1.2 to name a few).

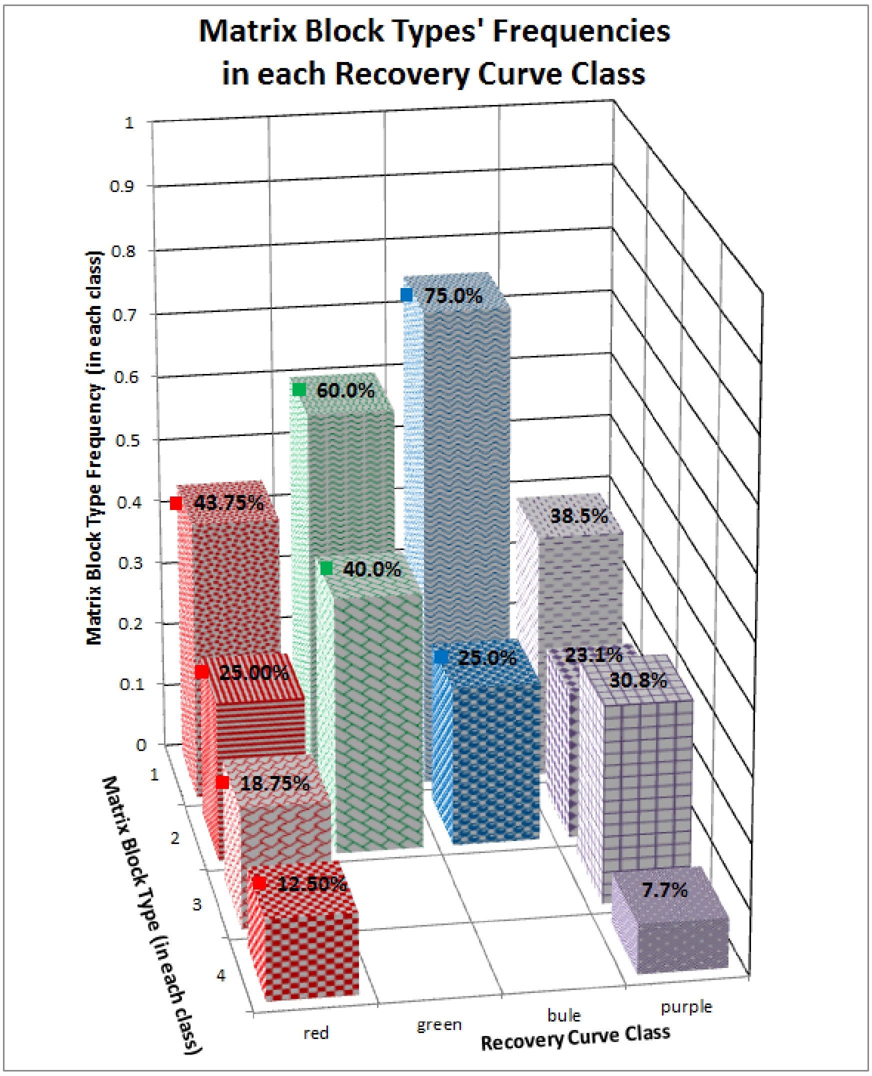


Figure 4.12: Matrix Block IDs' Frequencies for Figure 4.5

The simulation cell displayed in Figure 4.5, belongs to recovery curve region 1 in Figure 4.11. Each color indicates a common matrix block class; so it displays three matrix block classes in the recovery curve region of the simulation cell: red, green and blue with the frequency of about 40%, 25% and 35% respectively as shown in Figure 4.11 (with 0% of the purple matrix block class which exists in other recovery curve regions in the reservoir).

In Figure 4.5 each matrix block class, consists of different matrix block IDs (illustrated with different textures for each color). The frequencies of each matrix block ID in each matrix block class (in the reservoir) is displayed in Figure 4.12 (with different textures for each color representing a matrix block ID for that matrix block class). Figure 4.12 shows that each one of the four matrix block classes consists of different matrix block IDs (the red and purple have four and the green and blue matrix block classes have two matrix block IDs respectively) each with a known frequency (which sum up to 100%) for the matrix block class.

4.3.5.1 Summing up the matrix block recoveries

Consider the simulation cell displayed in Figure 4.1. As it can be observed, the cell consists of many matrix blocks of different IDs. The production from the matrix simulation cell during each time step, clearly is the summation of the productions of the matrix blocks. Since it is assumed that the matrix-fracture transfer of matrix blocks in each time step is independent of the other matrix blocks in the simulation cell, their production can be summed up for that time step to form the total transfer for the matrix simulation cell in that time step:

$$q_{mf}^{cell}(t) = \sum_{blocks} q_{mf}^{I}(t) \cdot f_I \quad , \tag{4.1}$$

where $q^I{}_{mf}$ is the transfer term and f_I is the frequency of occurrence for the matrix block I in the simulation cell. The transfer term can be calculated from the recovery curve for the matrix block as described in Section 3.8 and Equation 3.21 which is the method used and proposed here. It is however, also possible to calculate it from the conventional transfer function (Equation 2.8 with the shape factor from Equation 2.9 or Equation 2.10).

Chapter 5

Recovery Curve Approach to Simulation of NFRs

5.1 Introduction

In this chapter, the workflow for the new approach of using recovery curves for calculating the matrix-fracture transfer in the dual-porosity concept is explained in details.

Section 5.2.1 describes the detailed theoretical workflow to make the necessary preparations for the presented approach and Section 5.2.2 describes how to use these prepared items in a conventional simulator to benefit from the advantages of the additional accuracy that can be achieved by using recovery curves to calculate the matrix-fracture transfer. In Section 5.3, the steps to practice the recovery curve analysis in the industry are presented (so that this concept is not only limited to theory). Finally, in Section 5.4 the advantages and limitations of this new approach are mentioned.

5.2 The Workflow

The workflow is separated in two parts: pre-simulation (which is defining recovery curve regions and their distribution) and simulation (which is using the region recovery curves, to calculate the matrix-fracture transfer in the actual simulation):

5.2.1 Pre-simulation

It is obvious that for using recovery curves in the simulation, at first the matrix block IDs should be identified and their recovery curves should exist. Section 5.2.1.1 suggests how the matrix block IDs could be identified and Section 5.2.1.2 presents some of methods that can be used to prepare the recovery curves for the matrix block IDs. These curves, are used to define matrix block classes (Section 5.2.1.3), identification of recovery curve regions (Section 5.2.1.4) and to calculate the recovery curve for each region (Section 5.2.1.4.1) which will be used in the simulation step (Section 5.2.2).

5.2.1.1 Identifying Matrix Blocks and Their Distribution

As explained in Section 3.6, the matrix block shapes can be known from any source including outcrop investigation and/or discrete fracture models. Once the matrix block shapes are identified (i.e. each matrix block shape is numbered/named), the frequency of appearance of each matrix block ID in different parts of the reservoir can be considered to have the distribution of the matrix block IDs as well.

5.2.1.2 Generating Recovery Curves for Matrix Block IDs

The recovery curves for matrix block IDs can be from any source (as long as they represent the matrix-fracture transfer process under the specified set of conditions). In this section, three of such methods are briefly mentioned.

5.2.1.2.1 Laboratory Measurement

In order to measure the recovery curve for a certain matrix block under a certain set of conditions the matrix block should be cleaned and saturated with oil. Then the desired conditions are imposed in the laboratory apparatus and the recovery curve is created by plotting the collected oil versus time as explained in Appendix C.

Note that the same procedure can be repeated for any conditions e.g. under different pressures, fluids (if the fluids' type varies in the reservoir) or even applying a pressure gradient to the surrounding fluid to simulate viscous displacement (if it is expected that this happens in the reservoir). Each combination of the driving forces, generates a different curve which should be used for the matrix block ID during the simulation run, *if it undergoes the same condition*. In other words, it can be expected that the same matrix block ID exists in different parts of the reservoir: in one place it can be in the water-invaded zone while a matrix block of the same ID in some other part of the reservoir is in the gas-invaded zone; therefore, each of them will be depleted on different processes and from different recovery curves that can be pre-measured in the laboratory under their own similar condition. Hence for each block ID, different curves can exist for different conditions.

Also note that the orientation of the block *can* affect the gravitational driving force if the difference between the length of the sides of the matrix block is large. If block orientation affects the recovery trend much, each orientation should be regarded a different matrix block ID with its own recovery curve, and will be treated completely independently from the rest of orientations. Figure 5.1 shows one matrix block in two orientations (horizontal in red and vertical in blue). Figure 5.2 shows the recovery curves for each of these orientations. Since it is an isotropic matrix block with different orientations, the shape factor and pore volume is the same for both cases. The gravitational force is changed (as a result of different effective heights) which is the only factor responsible for the difference in the recovery curves. Therefore, when categorizing the matrix block IDs, these different orientations should be considered as two different matrix block IDs as if they were two completely different-looking blocks.

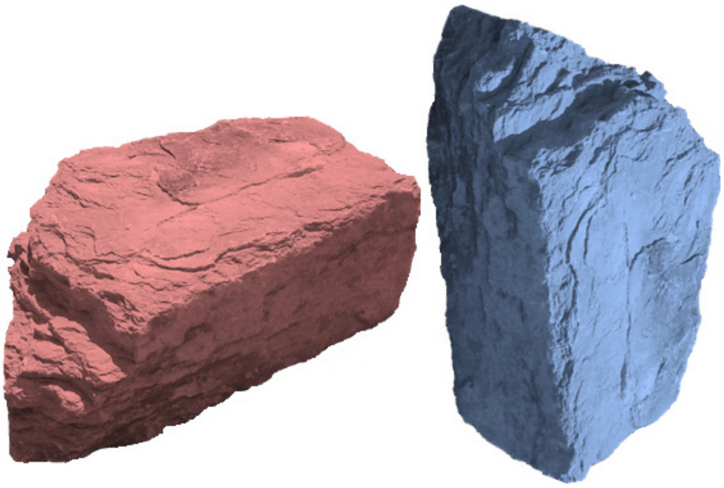


Figure 5.1: The same matrix block in different orientations

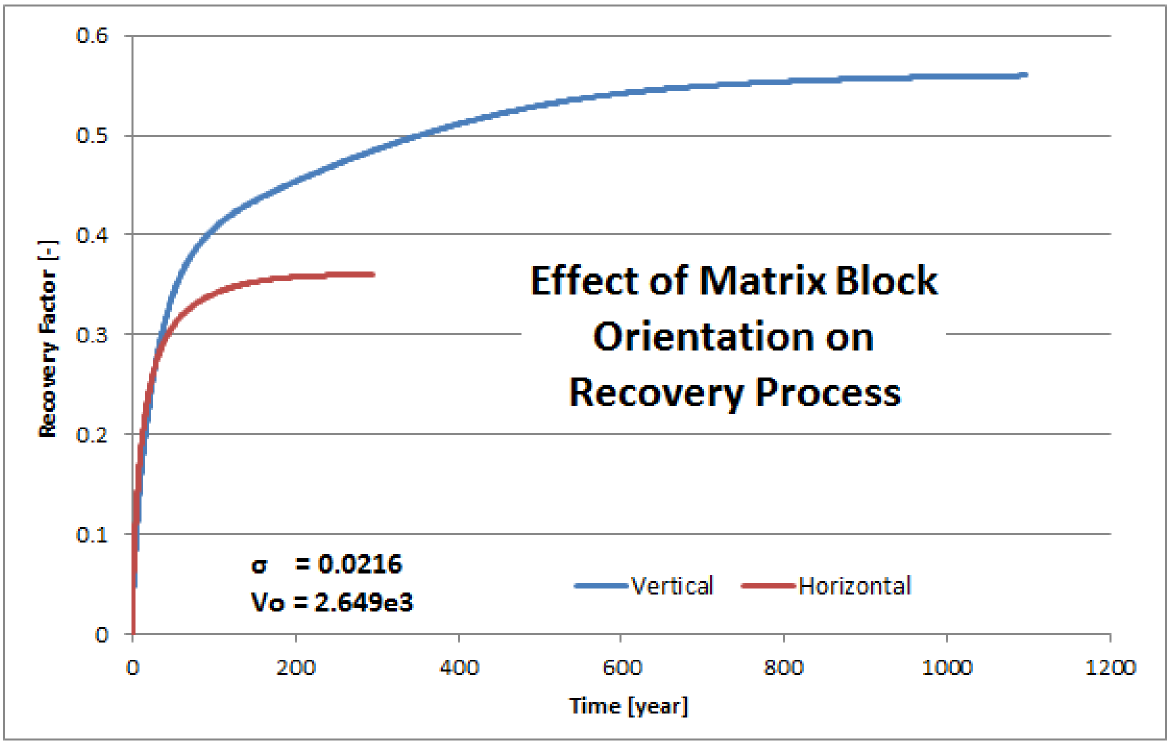


Figure 5.2: Recovery Curves of the matrix block IDs in Figure 5.1

5.2.1.2.2 Simulating Each Matrix Block ID

Since it is impossible to measure the recovery curve for all matrix block IDs in the laboratory (for all the possible conditions they might undergo throughout the reservoir during the course of simulation), it is more realistic to model the same laboratory experiment using a conventional single-porosity commercial reservoir simulator. It is possible to create the matrix block models of any shapes in any coordinate system exactly in the same manner as a full-scale reservoir is modelled.

A simulation model of the actual matrix block ID (with a grid representing it) can be created using any appropriate reservoir simulator. Different applications of small-scale single matrix block models can be found in the literature[20],[33],[41],[54],[89]. The recovery curve can be created from the simulation output of such models under the imposed boundary conditions. The shape factor can be measured as explained in Section 3.3 and the dimensionless time conversion factor from Equation 3.26.

5.2.1.2.3 Single Matrix Block Analysis

Since it is absolutely impractical to create small-scale reservoir models for every individual matrix block ID that exists in a reservoir, a new *standard* approach is presented and developed which can be used readily for any matrix block ID from an existing reservoir model. As explained in Section 3.4 and Appendix D in details, the new Single Matrix Block Analysis concept and tool can be utilized to use the shape factor of the matrix block ID (which is calculated as described in Section 3.3) from an existing reservoir model as well as the matrix/fracture rock and fluid types and the driving forces acting on the block to extract matrix oil to the surrounding fracture to output the recovery factor versus both time and the dimensionless time for the matrix block ID under the specified conditions.

5.2.1.3 Classifying Recovery Curves

After the recovery curves of all matrix block IDs (for the specified drive mechanisms) are known, their recovery curves (for every driving force) are scaled on the time axis as described in Section 4.3.2 to have the *scaled recovery curves*. All the matrix block IDs having the same scaled recovery curve, are classified as one *matrix block class* (as long as the driving mechanism is also the same for all of the curves). As displayed in Figure 4.8 and Figure 4.10 the dimensionless time conversion factor, combines several recovery curves (respectively in Figure 4.7 and Figure 4.9) as a single scaled recovery curve. It is necessary to keep the frequency of occurrence of each matrix block ID in each matrix block class. It is sufficient to tabulate the frequency of dimensionless time conversion factors of each matrix block ID in the matrix block class. Figure 5.3 shows the scaled matrix block class and Table 5.1 displays the dimensionless time conversion factor frequencies for the "red" matrix block class in Figure 4.12. For the other matrix block classes, similar tables exist as well.

Table 5.1: Defining a Matrix Block Class

Matrix Block ID	*Dimensionless Time Conversion Factor*	*Frequency [%]*
1	7.151476	43.75
2	2.860590	25.00
3	20.369980	18.75
4	1.220978	12.50

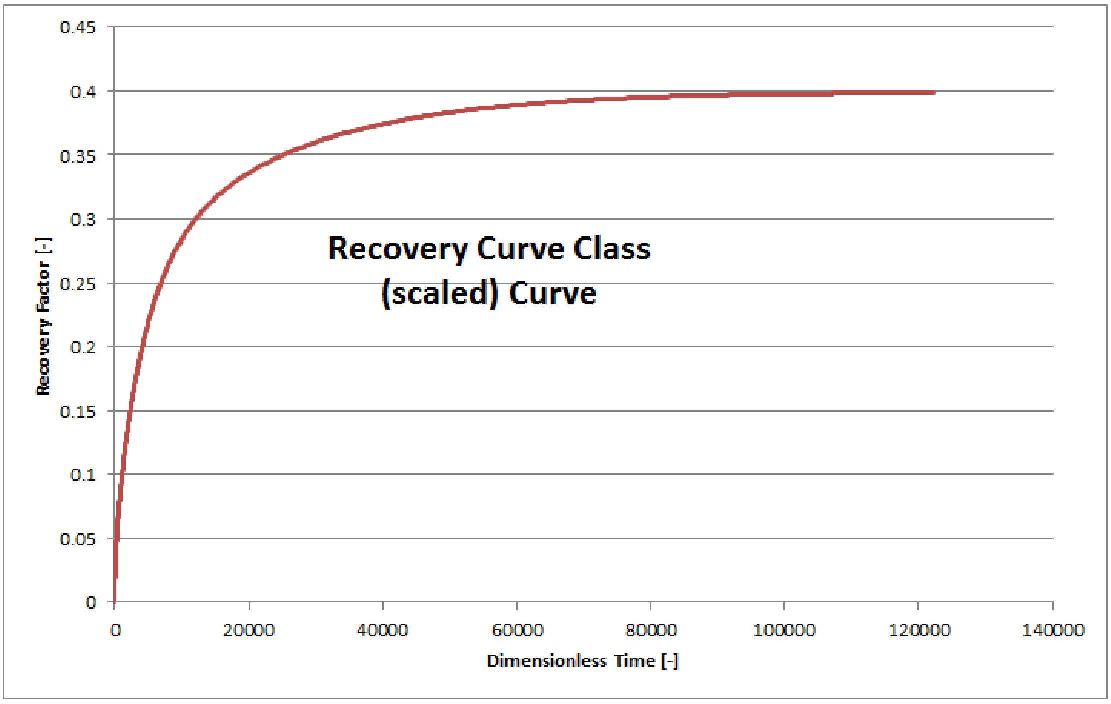


Figure 5.3: Matrix Block Class Scaled Recovery Curve

5.2.1.3.1 Calculating Class Recovery Curves

After preparing the scaled recovery curve (vs. dimensionless time) and defining the table pairs for each class (e.g. Figure 5.3 and Table 5.1), the recovery curve (vs. time) for the class can be easily calculated in the following steps:

1. Restoring the matrix block ID recovery curves from the scaled class recovery curve, by dividing the dimensionless time value of the matrix block class by the dimensionless time conversion factor for each matrix block ID. This step produces as many recovery curves (vs. time) as the number of matrix block IDs.
   (The thin colored lines in Figure 5.4 are all created by dividing the time value from the class recovery curve of Figure 5.3 by the second column values of Table 5.1.)
2. Then at every point in time, the values from each matrix block ID's recovery curve, is multiplied by its frequency (the third column of Table 5.1) and summed up to calculate the lumped class recovery curve (thick dashed black line in Figure 5.4).
   (See also Section 3.9 for more details about lumping recovery curves.)

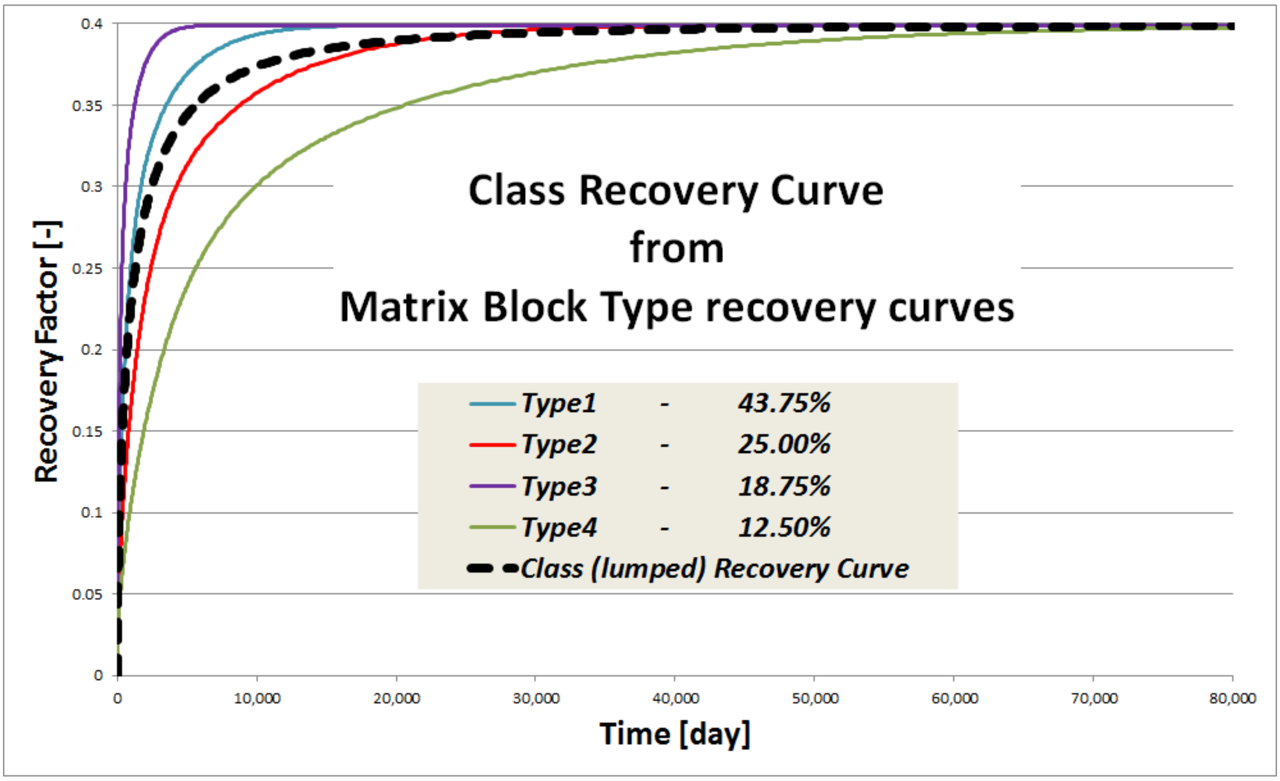


Figure 5.4: Calculating Class Recovery Curve

5.2.1.4 Defining Recovery Curve Regions

After the matrix block classes are set up, they should be mapped on the full reservoir (based on the gathered data of the matrix block IDs that exist at every point in the reservoir and the matrix block class that they belongs to). It will become clear that the matrix block classes are distributed with different frequencies in different regions of the reservoir. Based on these different frequencies, the reservoir can be split into different recovery curve regions.

Figure 5.5 is an example of a reservoir containing three recovery curve regions and their distribution in the reservoir. Each region consists of a known frequency of matrix block classes which was shown in Figure 4.11 and tabulated in Table 5.2. The frequency of each matrix block class is the number of matrix block IDs in a recovery curve region belonging to it, divided by the total number of matrix block IDs in the recovery curve region.

Different recovery curves can be assigned to a matrix block class for different conditions; therefore, each recovery curve region, too, can have more than one curve for different driving mechanisms and conditions. The region recovery curve for every set of conditions, is prepared from the class recovery curves (of that same condition set) as described in Section 5.2.1.4.1 and all the curves plus the condition of each one, define the recovery curve regions.

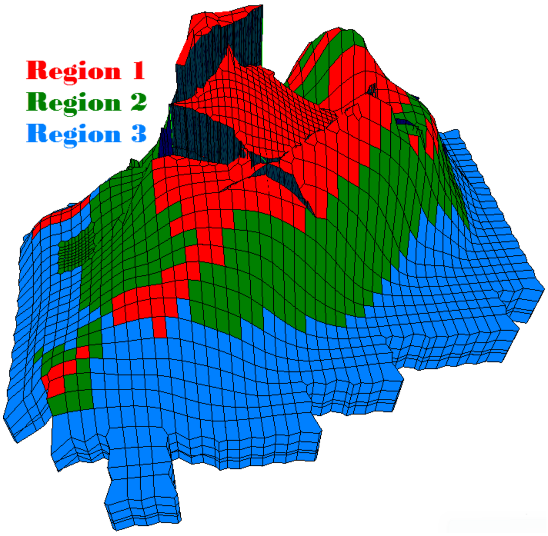


Figure 5.5: Recovery Curve Regions

Table 5.2: Defining Recovery Curve Regions

Class:	***1 - Purple***	***2 - Blue***	***3 - Green***	***4 - Red***
Region 1:	00.0%	35.0%	25.0%	40.0%
Region 2:	20.0%	25.0%	15.0%	30.0%
Region 3:	45.0%	10.0%	10.0%	35.0%

5.2.1.4.1 Calculating Region Recovery Curves

After defining the recovery curve regions and calculating the distribution of each matrix block class in each region (e.g. Table 5.2), the recovery curve (vs. time) for each region can be easily calculated with the same procedure described in Section 3.9 for lumping recovery curves with known frequencies.

Figure 5.6 plots the class recoveries from water imbibition drive for the four recovery classes in the example shown in Figure 4.11 as solid lines. The curves are lumped (multiplying the recovery factor value of each class by its frequency and summing them up for all classes to have the region recovery factor for every time point) with the frequencies shown in Table 5.2 to calculate the region recovery curves for each of the three regions (shown in thick broken lines). It can be observed in Figure 5.6 that the same set of matrix block classes form different region recovery curves just by different frequencies. Note that in this example, in recovery curve region 1, no "purple" matrix block class exists.

It is worthwhile to remind that all the recovery curves in Figure 5.6 (both class recovery curves and region recovery curves) are for a certain set of conditions and driving mechanism (e.g. here for water drive). For each drive mechanism, a similar curve should be assigned to each recovery curve region.

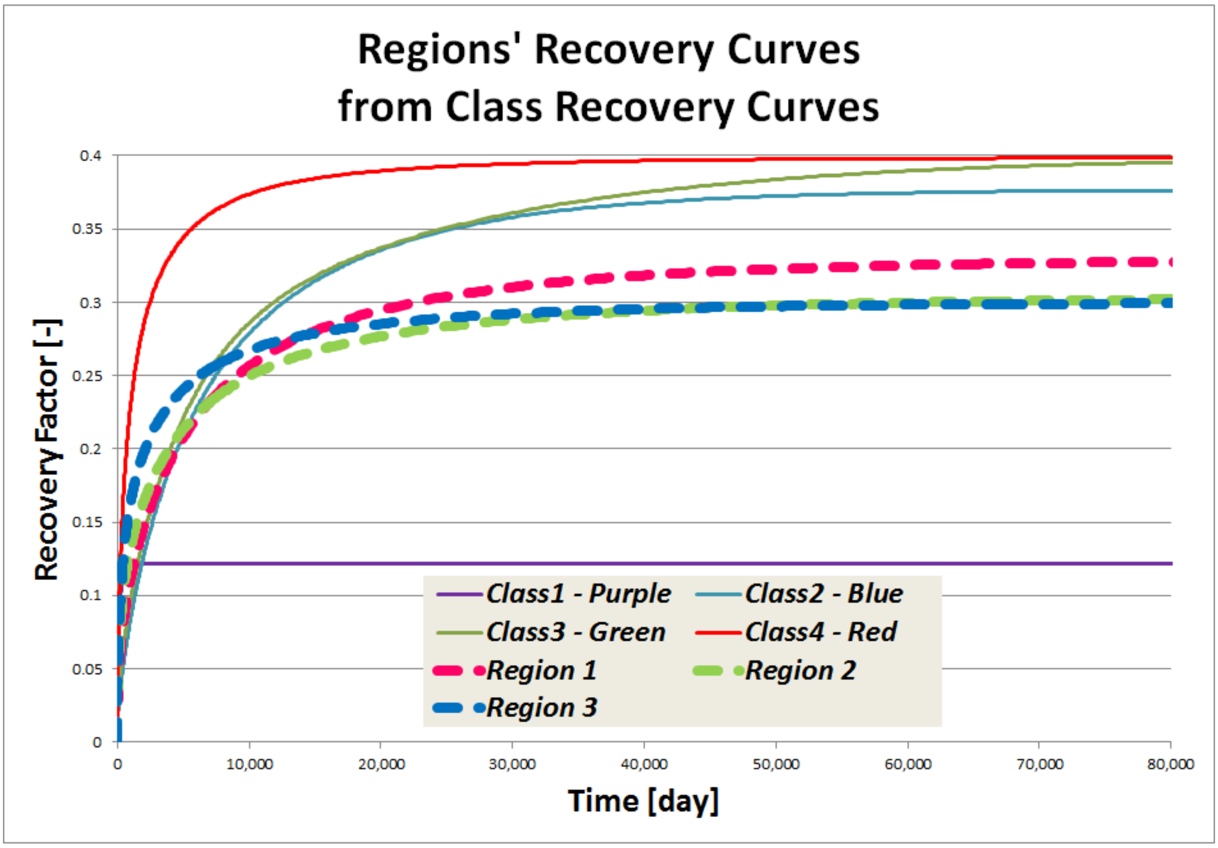


Figure 5.6: Regions' Recovery Curves from Class Recovery Curves

5.2.2 Simulation

After performing all the pre-simulation steps described in Section 5.2.1, the following information is available to be used as additional input for simulation of NFRs using recovery curves:

- Definition of recovery curve regions: recovery curve (vs. time) for every drive mechanism and set of conditions for every region.

- Recovery curve region distribution in the reservoir: to every simulation cell, one (and only one) recovery curve region is assigned.

These additional input data are used to calculate the matrix-fracture transfer term, instead of Equation 2.8 in the conventional dual-porosity simulation *if the appropriate recovery curve exists for the conditions the cell undergoes in the run*. If the recovery curve does not exist for the current conditions of the simulation cell, still the conventional transfer function (Equation 2.8) will be used until the condition of the cell changes to match the condition of an available recovery curve, after which the matching recovery curve will be used as explain in the following section.

5.2.2.1 Description of the new approach

As already mentioned, the new approach is an improvement to the conventional dual-porosity formulation explained in Section 2.4.2. The continuity equations are the same as Equation 2.6 and Equation 2.7 for the fracture and the matrix domains for every simulation cell (in both fracture and matrix domains respectively).

For calculation of the matrix-fracture transfer term, q_{cmf}, during the course of simulation for every simulation cell, the following steps are performed:

- The actual conditions (pressure, active drive mechanisms etc.) of the cell (at each time step) are identified.
- If the recovery curve region that the cell belongs to, does not contain a curve for the actual conditions of the block, the matrix-fracture transfer term is calculated from the conventional method, Equation 2.8, for that time step.
- If the recovery curve region that the cell belongs to, contains the curve for the actual conditions of the block, the matrix-fracture transfer term is calculated with the procedure described in Section 3.8:
  - The actual recovery factor of the block is calculated from Equation 3.20.
  - The volumetric matrix-fracture transfer rate during the current time step is calculated from Equation 3.21.

Everything else is handled exactly in the same manner as the conventional approach. The advantages and limitations of the new approach, are explained later in Section 5.4.

5.3 Practical Steps for Full Field Simulation

The theoretical workflow explained in Section 5.2 assumes the situation where all the matrix block IDs are known and the recovery curves for all of them is generated. This procedure, although could be possible, still has too much of pre-simulating work which might make the procedure impractical. In this section, an equally acceptable practical workflow which can be used in the industry is presented as well.

The matrix block IDs can still be identified from the discrete fracture model as explained in Section 5.2.1.1. However, in the practical step, instead of generating the recovery curve for every single matrix block ID, they are manually classified considering the dimensionless time conversion factor formula Equation 3.26. The rule of thumb for manually classifying the matrix block IDs (just by their appearance and location in the reservoir) would be:

> "*All matrix block IDs with the same rock and fluid IDs and approximately similar effective heights and not so-variant initial conditions can be readily counted as one matrix block class*".

Note that in this definition, the matrix block IDs can have different effective permeabilities, porosities and more importantly, *shape factors* and still be considered in the same class. The reason is that the dimensionless time conversion factor, Equation 3.26, uses all these parameters (even the initial oil viscosity and compressibility) as the scaling parameter. Since the Darcy's flow equation is used for derivation of the proposed dimensionless time conversion factor, the basic flow theories apply for scaling. For example, if the porosity is higher or the apparent permeability is less, the block takes longer to deplete and the k_a/ϕ ratio scales the time access accordingly (see for instance Figure 4.7 and Figure 4.8).

5.3.1 Pre-Simulation

The interesting part which needs more consideration, is the occurrence of the shape factor in the derived scaling factor for the time axis. This means that the matrix block IDs of different shapes would have the same scaled recovery curve (given that the driving forces are similar). This is the same fact which was discussed in Section 3.2.3 as "Shape vs. Shape Factor". This gives the great practical advantage that reduces the amount of pre-simulation workload several times, and as a result, the practical steps would be the following:

1. Identifying matrix block IDs (as described in Section 5.2.1.1).
2. Calculating the shape factor for each matrix block ID (Section 3.3).
3. Calculating the dimensionless time conversion factor for each matrix block ID (Section 3.10).
4. Gathering all matrix block IDs which fit into the rule of thumb explained in Section 5.3 as one matrix block class.
5. Choosing one matrix block ID from each matrix block class to calculate the scaled recovery curve (vs. dimensionless time) for the whole class.
6. Scaling the dimensionless time axis from the previous step, back to the time axis for all the matrix block IDs in the class. This step is nothing more than a simple division of the dimensionless time axis values by the dimensionless time conversion factor for each matrix block ID.
7. Calculating the class recovery curve, in the same manner as described in Section 5.2.1.3.1.
8. Calculating the region recovery curves, in the same manner as described in Section 5.2.1.4.1.

As it can be observed, all the practical steps are identical to the theoretical workflow except for steps 4-6. In the practical approach, it is not needed to measure/simulate the recovery curves for every single matrix block IDs; but instead, only one block in the matrix block class is selected to calculate the scaled recovery curve of the class. From this scaled recovery curve, the recovery curves of all the other matrix block IDs in that class is simply calculated. This is a very significant *advantage of using dimensionless time conversion factor and scaling recovery curves* which is promoted here and effectively reduces the workload several times.

5.3.2 Simulation

Since it is impossible to generate region recovery curves for every possible set of conditions that all the cells of that region might undergo during the run, the recovery curves are used in a practically acceptable threshold above and/or below the actual conditions under which they were created. For instance, if the pressure goes (to a certain limit) below or above the actual pressure in which the recovery curve was created, still the same curve would be used to calculate the matrix-fracture transfer. The same is true for the fracture fluid saturation, e.g. if the curve is created considering that the fracture is fully water-saturated, it can be practically accepted that the same curve still can be applied if the fracture fluid saturation goes below that down to a certain threshold value (e.g. until saturation of 85%). However, if the difference is larger than the acceptable threshold, and no recovery curve exists for that condition for the region (e.g. fracture water saturation is 50%), the fracture-transfer term will be conventionally calculated.

5.4 Advantages and Limitations

The advantages of the developed single matrix block (SMB) analysis and the recovery curve (RC) approach over discrete fracture network (DFN) approach and the conventional dual-porosity sugar cube (SC) approach are as follows:

- In SC the homogenization of the matrix blocks in the simulation cell is too high to be either physically or practically acceptable. In DFN, the level of details is too high to be industrially applicable; even if DFN is used for upscaling the shape factor distribution to be used for large reservoir models, still a similar over-homogenization and using a single shape factor value for an entire simulation cell is inevitable. RC approach brings the results of a detailed study on the matrix blocks in the large scale of the simulation cell with superposition of the effects of each matrix block ID it contains in time. Theoretically, RC needs no homogenization, yet it can still be used to simulate any large-scale reservoir that SC can simulate.
- The drive mechanisms in RC can be studied in detail with no theoretical simplifications in laboratory or using mathematical models and included in the recovery curve for each matrix block. This is simply not feasible in SC since always some simplifications has to be considered to be able to calculate the effect of the combination of the acting drive mechanisms in the transfer function. DFN also fails in the sense that it tries to calculate all the interaction of the active drive mechanisms for every matrix block during the course of the simulation which is too many calculations for an industrially applicable solution.
- SMB can be used to create recovery curves for matrix blocks in different conditions to be used for creation of the lumped class recovery curves and the recovery curve regions which will be used in the dual-porosity calculations. All these steps are available to be used on any dual-porosity reservoir on a commercial reservoir simulator[45].
- The additional level of accuracy brought by RC, allows better simulation results and allows using recovery curves as a better history matching parameter.

The following items were outside the planned scope of this work and were not considered in the described recovery curve approach devised and improved in this work:

- the effects of hysteresis on the recovery curve calculation and the simulation with recovery curve,
- endpoint scaling,
- compositional fluid description, and
- matching the recovery curves based on historical data.

The theoretical recovery curve approach which was devised and improved in this work was implemented in ***PRS***[45], an industrial commercial reservoir simulator to the following extent:

- The matrix-matrix transfer (dual-permeability) is calculated during each time-step as in SC, but at the beginning of the next time-step it is assumed that all the recovery is a result of matrix-fracture transfer. This assumption was tested on a real reservoir with practically no difference with the complete dual-permeability case.
- The implementation of the recovery curves does not consider pressure changes. In practice, it was tested that an average under-saturated recovery curve for operating pressures above the bubble-point pressure and a few saturated recovery curves for the pressures below the bubble-point pressure can be used, but switching the recovery curves while simulation creates a shock that goes along with the pressure drop shock and forces the run to be calculated with very small time-steps. Therefore, this option was not implemented and always one curve (per drive mechanism) is used.
- The drive mechanisms are identified based on the saturation of the fracture cell:
  - if the water saturation in the fracture cell is above 85%, the drive mechanism is assumed to be water imbibition;
  - if the gas saturation in the fracture cell is above 85%, the drive mechanism is assumed to be gas gravity drainage;
  - for any other transitional state between the mentioned states, the conventional transfer formula is used to calculate the matrix-fracture transfer.
- If the drive mechanism changes during the run, the actual drive's recovery curve will be used and all the previous recovery would be considered to be from this drive (e.g. if a fully water-filled fracture cell becomes gas-filled during the simulation run, no longer the water RC is used and instead the gas RC will be used for the future and the starting point on the gas recovery curve, is the cell's actual recovery factor exactly as if the block were being depleted from the beginning with the gas).

Chapter 6

Verification and Case Study

6.1 Introduction

In this chapter the proposed method of using recovery curves to simulate naturally fractured reservoirs is verified in a case study and by comparison with the conventional run for a column model with the same properties as in a real reservoir model.

6.2 The Column Model Description

In order to verify and demonstrate the applicability of the proposed recovery curve method, a dual-porosity column model was set up. The reason that a column model was used is that in this model, every matrix cell is connected only to one fracture cell and therefore, the effect of matrix-fracture is emphasized and can be studied for verification of the recovery method for calculating the matrix-fracture transfer term.

The model consists of a column of 700 one-meter-thick dual-porosity cells and a fifty-meter-thick single-porosity aquifer cell at the bottom. The cells are 100 meters on each side. The PVT and rock data are taken from a real field. Some of properties of the model are listed in Table 6.1. The capillary pressures and relative permeabilities are plotted in Figure 6.1 and Figure 6.2 respectively for oil/water and oil/gas systems.

A constant-pressure aquifer is attached to the bottom cell and a liquid producer well is completed on the top layer in the fracture, producing with a constant rate of 50 standard cubic meters per day. No maximum watercut is specified so that the well continues to produce even if it is producing 100% water; the reason is that in this work the main focus point is the trend of entrance of water to matrix cells; and the goal is to allow the bottom half of the model to be nearly completely depleted which needs more time than the time of water breakthrough to the well.

The model is dual-porosity to allow concluding that any entered water in the matrix must have come from the fracture (and not from a matrix neighbor). In this way, the matrix-fracture transfer in every simulation cell can be accurately monitored (and used for verification).

This base model is used to create three different cases starting from a homogeneous one to a complex case and for each case, the conventional simulation is performed, necessary recovery curves are created and used in a recovery curve run and finally the results of the recovery curve simulation run are compared against the conventional run.

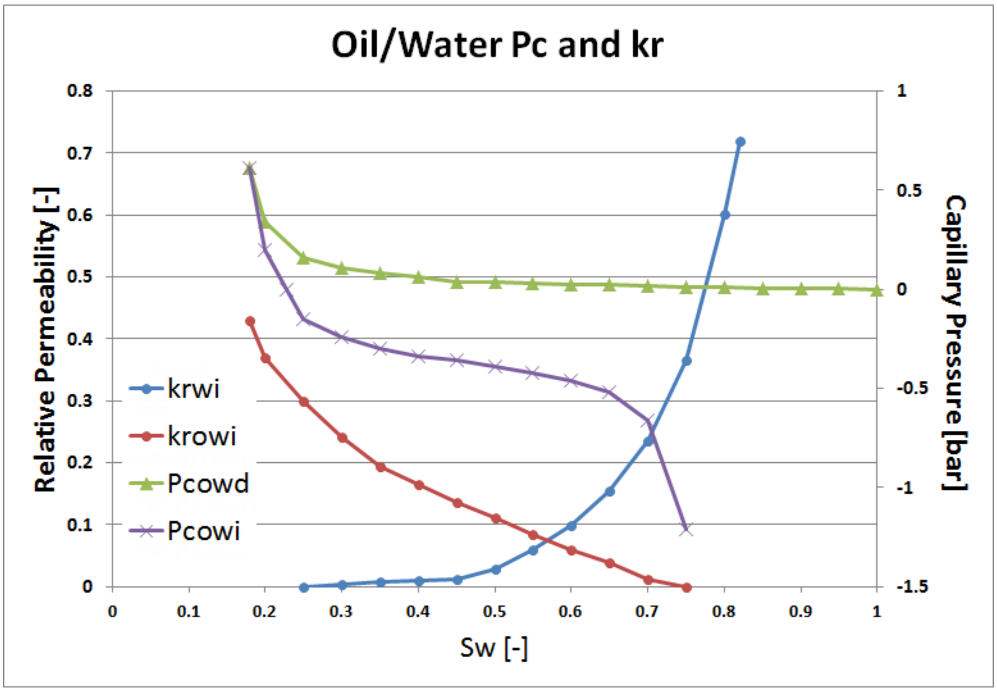


Figure 6.1: Oil/Water capillary pressures and relative permeabilities

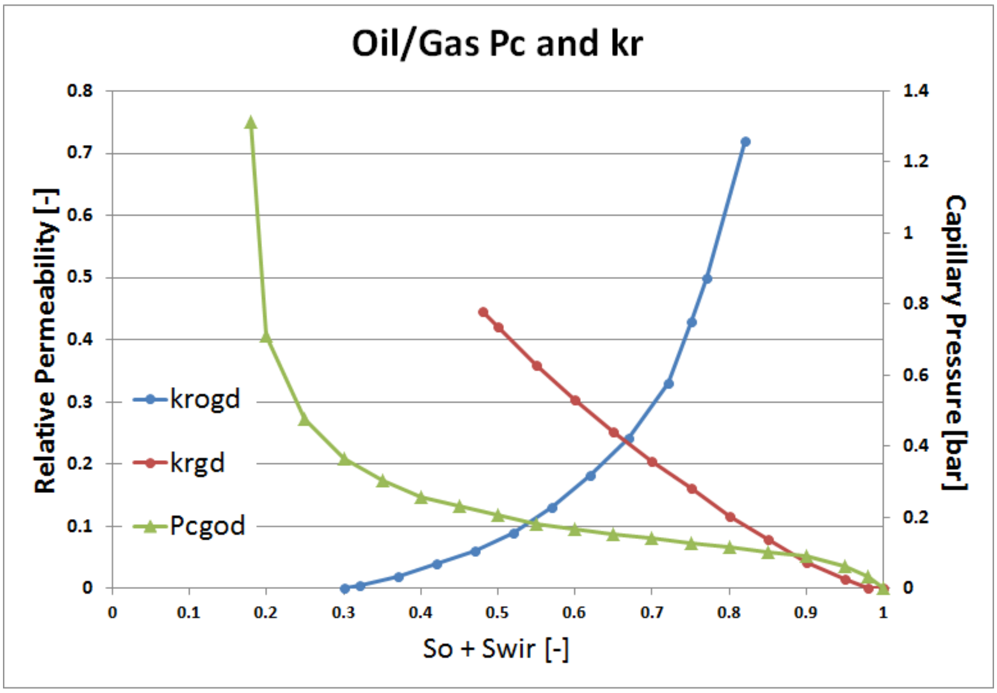


Figure 6.2: Oil/gas capillary pressure and relative permeabilities

Table 6.1: Model Properties

Water Density	1037.491 [kg/m^3]
Oil Density	786.487 [kg/m^3]
Gas Density	1.025 [kg/m^3]
Water Viscosity	0.31 [cp]
Water Compressibility	4.51E-4 [bar^{-1}]
Oil Viscosity @Pb	1.728 [cp]
Oil Formation Volume Factor @Pb	1.723 [rm^3/sm^3]
Solution GOR @Pb	253.459 [sm^3/sm^3]
Bubble Point Pressure	139.0 [bar]
Water Oil Contact	1001.0 [m]
Initial Pressure at WOC	330.95 [bar]

6.3 *Case I* - Verification

Case I is a homogeneous case used for verification of the recovery curve approach. The idea is that a simulation cell in a conventional run can be monitored and its recovery curve can be extracted (see Section 3.7). The validity of the proposed method for calculation of the matrix-fracture transfer term using the recovery curve is verified if inputting the derived curve from the conventional run in the recovery curve run, gives similar results to the conventional.

The implemented procedure allows using the recovery curves from every individual cell in the model from the conventional run, in the recovery curve run by assigning a different recovery curve region to each cell. The results of the conventional and the recovery curve runs were identical: because the same matrix-fracture transfer from the conventional run for each cell was used as the input for the recovery curve run and since this step's result is too obvious, this step's results are not demonstrated here and instead, the results of one step further are discussed which validates the model even if the recovery of one cell from the conventional run is used and assigned to all cells for the recovery curve run.

In other words, instead of "outputting the recovery curves from *every matrix cell* in the conventional run and in the recovery curve run, assigning to each cell its own recovery curve", this time "the recovery curve from *one matrix cell* in the conventional run is used and assigned to all cells in the recovery curve run". All the grid properties are the same for all cells and are as listed in Table 6.2. The red curve plotted in Figure 6.3 is extracted from one cell during a conventional simulation.

Table 6.2: *Case I* - Properties

Property	***Matrix***	***Fracture***
Porosity [-]	0.2	0.01
Permeability [mD]	1.0 [2 2 0.25]	10000
Matrix Height [m]	15	-
Shape Factor [m^{-2}]	0.1	-

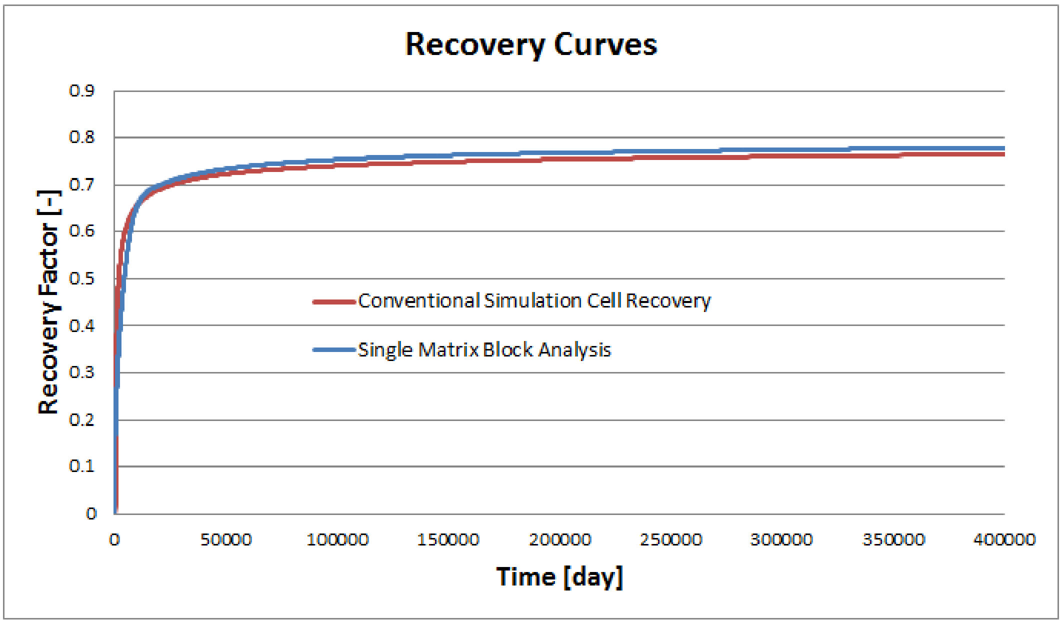


Figure 6.3: Recovery Curves used for the recovery curve simulations of *Case I* and *Case II* (red and blue respectively)

The red curve in Figure 6.3 was used as the recovery curve input for all cells in for the recovery curve run. It is worthwhile to mention that for every cell in the recovery curve run, as long as the water saturation in the fracture is less than the threshold of 85%, the matrix-fracture transfer term is calculated conventionally and after water enters the neighboring fracture cell, the matrix-fracture transfer term will be calculated as described in Section 3.8.

Figure 6.4 is the plot of the conventional run's water production rate (in dark blue), oil production rate (in dark green) and average reservoir pressure (in red) and the plot of the recovery curve run's water production rate (in light blue), oil production rate (in light green) and average reservoir pressure (in pink). As it is clear in Figure 6.4, the results are close to identical for the two runs: 1) the conventional run and 2) using the recovery curve from one cell in the conventional run and using it as input recovery curve for all cells in the recovery curve run.

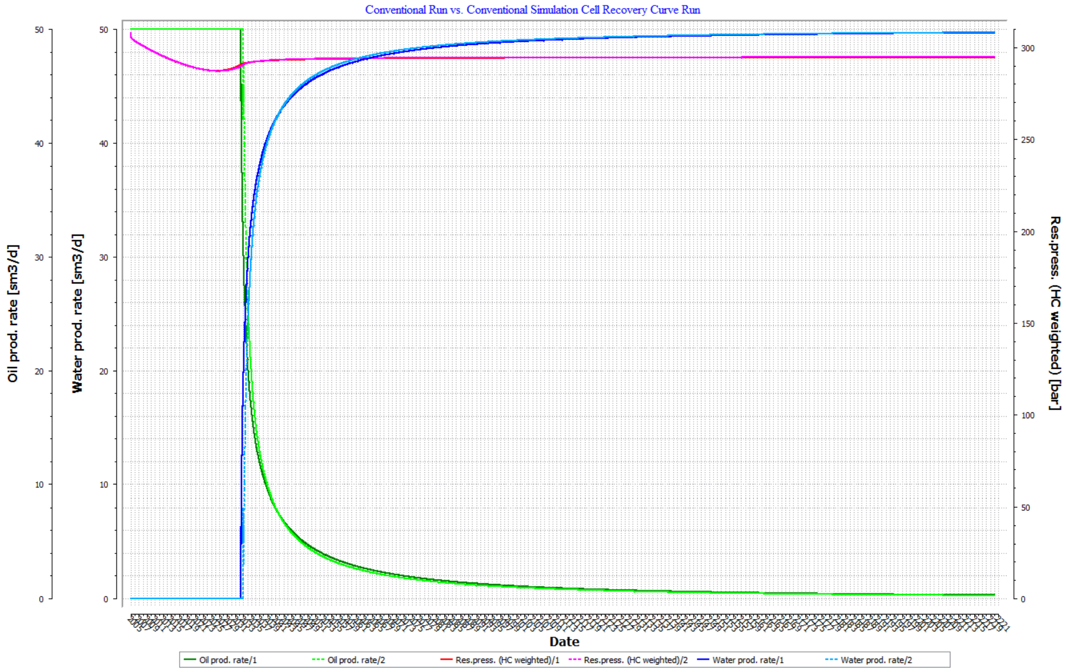


Figure 6.4: *Case I*: Reservoir Production Rates and Pressure

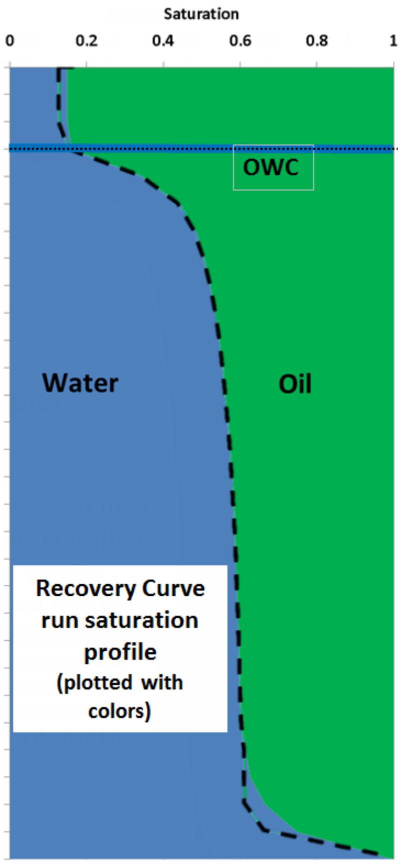


Figure 6.5: Vertical saturation distribution before water breakthrough for the conventional run (with dashed line) and the recovery curve run (in colors)

Figure 6.5 is the plot of vertical saturation distribution in both the conventional run (plotted with the dashed line) and the recovery curve run (plotted in colors) at the same point in time before the water breakthrough to the producing well. As it can be observed, not only the production and reservoir pressure are the identical for the two runs, but even the saturation distributions are.

This validates the proposed method and proves that the recovery curve method can be used as an alternative approach for calculation of the matrix-fracture transfer term in the conventional dual-porosity model.

6.4 *Case II* - SMBLOCK Analysis Curve

In the next step, a Single Matrix Block Analysis (see Section 3.4 and Appendix D) is performed to calculate the recovery curve from water imbibition for the conditions of the model (listed in Table 6.1 and Table 6.2). The result of the SMBLOCK Analysis is the blue curve in Figure 6.3. As it can be observed Figure 6.3, the recovery curve from the SMBLOCK Analysis *looks* very close to the cell recovery curve from conventional simulation.

Case II is identical to *Case I* except that for the recovery curve run, the recovery curve calculated from the SMBLOCK Analysis is used (instead of the recovery curve output from the conventional run). Figure 6.6 is the plot of the conventional run's water production rate (in dark blue), oil production rate (in dark green) and average reservoir pressure (in red) and the plot of the recovery curve run's water production rate (in light blue), oil production rate (in light green) and average reservoir pressure (in pink).

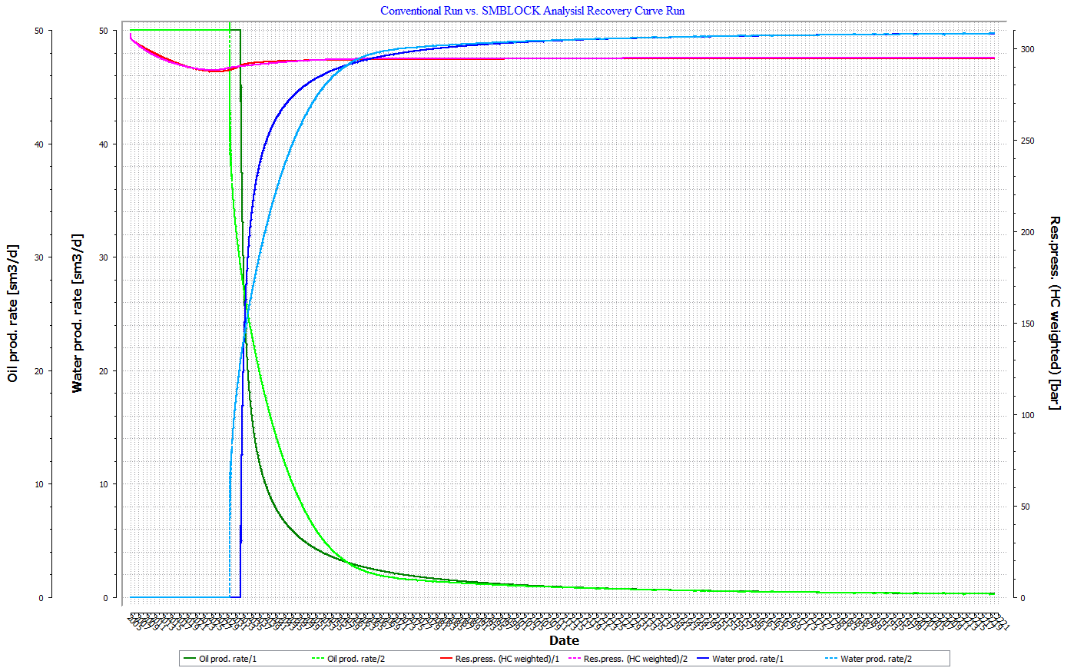


Figure 6.6: *Case II*: Reservoir Production Rates and Pressure

As it can be observed in Figure 6.6, although the results are still close enough, they are not as close to the conventional run as they were in *Case I*; the only difference in input data is that once the red and once the blue curve in Figure 6.3 (which look very similar to each other) are used. What should be kept in mind is the time scale of Figure 6.3 (which is about 11 centuries) to have the specified cell fully depleted while the actual simulation is performed to only about 2 centuries, meaning that none of the cells would be fully depleted in the simulation even after this long period.

6.4.1 Discussion

Figure 6.6 displays the simulation results from the two models. One is the conventional model and the other is the proposed method to use recovery curves. It is not possible in any ways to say if one result is *correct* or *more correct* just by looking at Figure 6.6. However, it is possible to argue that the difference comes from different methods of considering the drive mechanisms which are the only difference between the two curves in Figure 6.3.

As discussed in Section 3.4.8.3.1, the single matrix block analysis (Section 3.4) does not make the physically incorrect assumption of phase segregation in the matrix block and therefore can account for the gravitational and capillary forces more physically appropriately. Therefore, it is expected that the proposed recovery curve approach is more physically correct and closer to the reality than the conventional approach.

Additionally, in reality the amount of oil which enters the fracture, usually goes away much faster than the oil needs to come out from the matrix. In other words, it can be assumed that as soon as a droplet of oil comes out of the matrix to the water-filled fracture, it raises to the top leaving the matrix rock again in contact with a fully water-filled fracture (see Appendix C). But in the conventional approach, the fracture saturation is calculate only at the intervals of the time steps and the fracture is practically never fully water-saturated (as explained in Section C.2), while this fracture saturation is taken into account for calculation of the gravitational force (explained in Section E.4) which makes the results different from the more accurate and detailed calculation of the interaction of the forces that is considered in the recovery curve approach.

6.5 *Case III* - Multi-region Case

Case III is actually a transition step that was created between *Case II* and *Case IV* which has to be created because the conventional method is incapable of calculating the upcoming *Case IV*. Thus, here the model of *Case III* which is the basis for *Case IV* is explained.

For this case the data in Table 6.1 are still valid but instead of the homogeneous properties of Table 6.2, seven property regions are defined as in Table 6.3 with the distribution displayed in Figure 6.7.

Table 6.3: *Case III* - Properties

	PORO [-]	***PERM [D]***	***LZ [m]***	***Matrix Block ID Distribution***
Region 1	0.2	0.001	15	1
Region 2	0.5	0.025	15	2
Region 3	0.2	0.005	15	2
Region 4	0.3	0.002	15	3
Region 5	0.2	0.001	10	5 (the same as 1)
Region 6	0.5	0.025	10	5 (the same as 1)
Region 7	0.5	0.001	10	4

As can be seen in Table 6.3, each region has a *Matrix Block ID Distribution*. This represents the *matrix block*s which exist in each region. In other words, each simulation cell in these regions is expected to have *matrix block*s with the given distribution ID (from 1 to 5 where distribution ID, 5 is the identical to 1, except that the matrix block height is different).

For *Case III*, the weighed-averaged shape factor values listed in Table 6.4, represent each matrix block ID. Figure 6.8 is the plot of the conventional run's water production rate (in dark blue), oil production rate (in dark green) and average reservoir pressure (in red) and the plot of the recovery curve run's water production rate (in light blue), oil production rate (in light green) and average reservoir pressure (in pink). As it can be observed in Figure 6.8, the two runs still show similar trends, but the additional complexity has brought more difference between the conventional run and the recovery curve run in *Case III* comparing with *Case II*. More explanations about this difference will be given in *Case IV*.

Table 6.4: *Case III* - Matrix Block IDs

Matrix Block ID	***Weighted-Averaged Shape Factor [m^{-2}]***
ID 1	0.121
ID 2	0.693
ID 3	0.222
ID 4	0.507

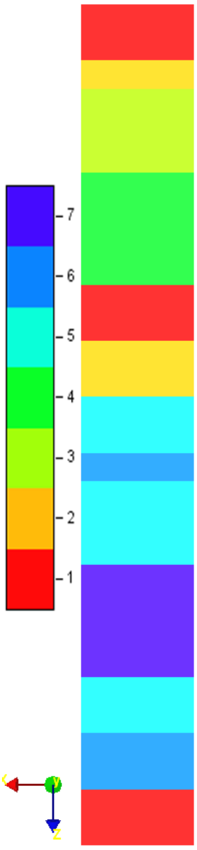


Figure 6.7: *Case III*: Property Regions’ Distribution

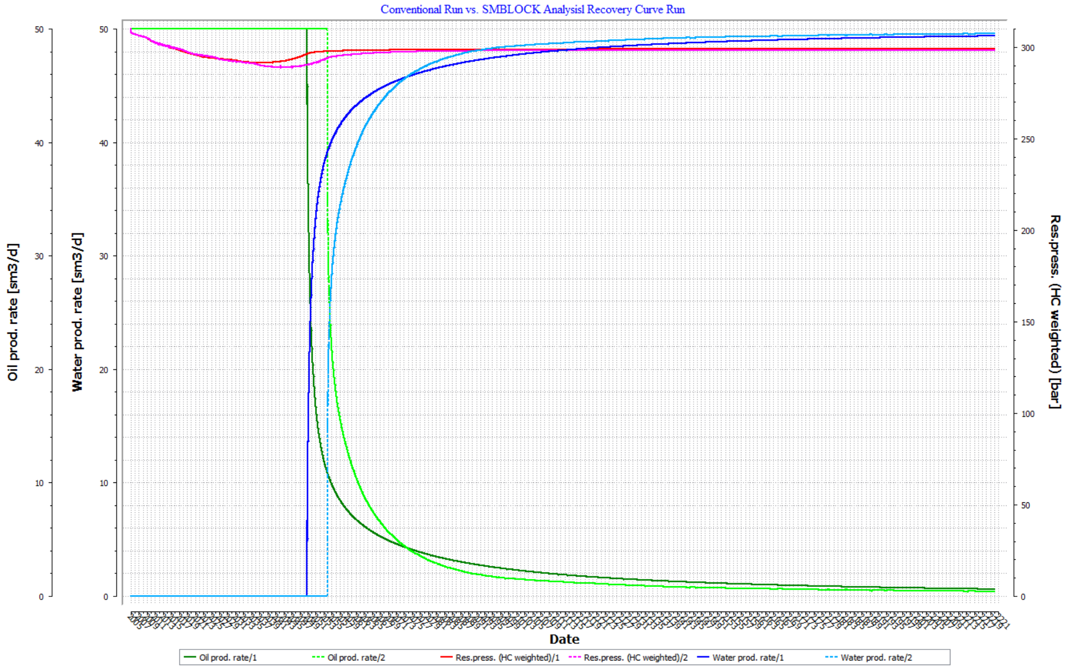


Figure 6.8: *Case III*: Reservoir Production Rates and Pressure

6.6 *Case IV* - Matrix Block ID Distribution

Case IV is the most complicated column model test that is presented. As explained, it is the same as *Case III* except that instead of using the averaged values for shape factors of Table 6.4 for the matrix block IDs of Table 6.3, the actual values listed in Table 6.5, Table 6.6, Table 6.7 and Table 6.8 are used for each region.

As it can be observed, for each matrix block ID of Table 6.3, a shape factor distribution (shape factor and frequency pair) is available as input. This means that each simulation cell in each region, contains different matrix block IDs with different shapes and trends of depletion under certain drive mechanism (which is always the case in reality).

The conventional approach assumes an average shape factor for each simulation cell, imposing that all the matrix blocks are identical; but the new recovery curve approach allows taking this physical reality into calculations (the differences were explained in details in Section 4.2 "Modeling Simulation Cell"; see Figure 4.2 and Figure 4.5 for the representation of the simulation cell in the conventional and the recovery curve approaches respectively for the physical model of Figure 4.1).

Table 6.5: *Case IV* - Matrix Block ID 1

Shape Factor [m^{-2}]	***Frequency [-]***
0.25	0.28
0.08	0.40
0.15	0.03
0.05	0.29

Table 6.6: *Case IV* - Matrix Block ID 2

Shape Factor [m^{-2}]	***Frequency [-]***
0.32	0.40
0.02	0.30
5.50	0.10
0.045	0.20

Table 6.7: *Case IV* - Matrix Block ID 3

Shape Factor [m^{-2}]	***Frequency [-]***
0.22	0.37
0.30	0.31
0.18	0.12
0.13	0.20

Table 6.8: *Case IV* - Matrix Block ID 4

Shape Factor [m^{-2}]	***Frequency [-]***
0.5863	0.59
0.35	0.17
0.44	0.14
0.39	0.10

Figure 6.9 is the recovery curve plots for Matrix Block ID 3 (Table 6.7). The recovery curves for each shape factor is respectively drawn in blue, green, red and purple lines. The recovery curve with the average shape factor, is plotted with black dotted line and the lumped (class) recovery curve (see Section 3.9) in black dashed line.

As it can be observed, the difference between the average shape factor r recovery curve and the lumped recovery curve here is even larger than for the two curves in Figure 6.3 (which was responsible for the differences in *Case II*).

Each individual matrix block ID curve (which represents the detailed production trend for that *matrix block*) has a direct influence on the final lumped cell recovery curve (exactly in the same way as it is in the reality). However, in the conventional method a completely different *hypothetical matrix block* is imagined to represent the average production trend of all the *matrix block*s which is shown in Figure 6.9 to be not the a correct assumption comparing to the physically correct lumped curve..

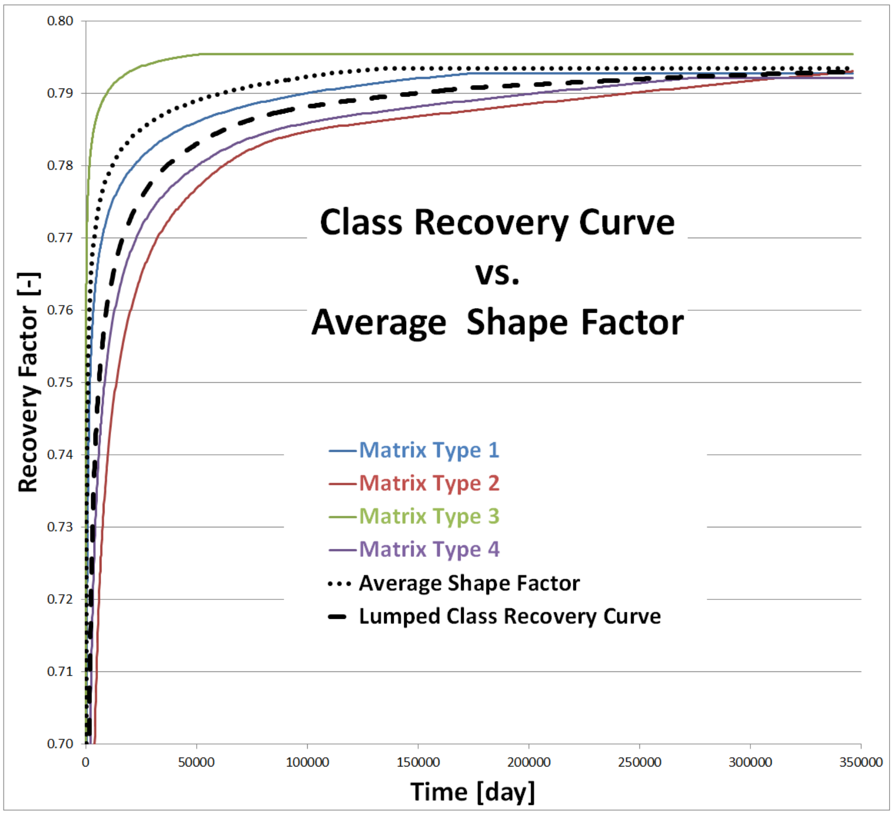


Figure 6.9: Recovery curves for Matrix Block ID 3

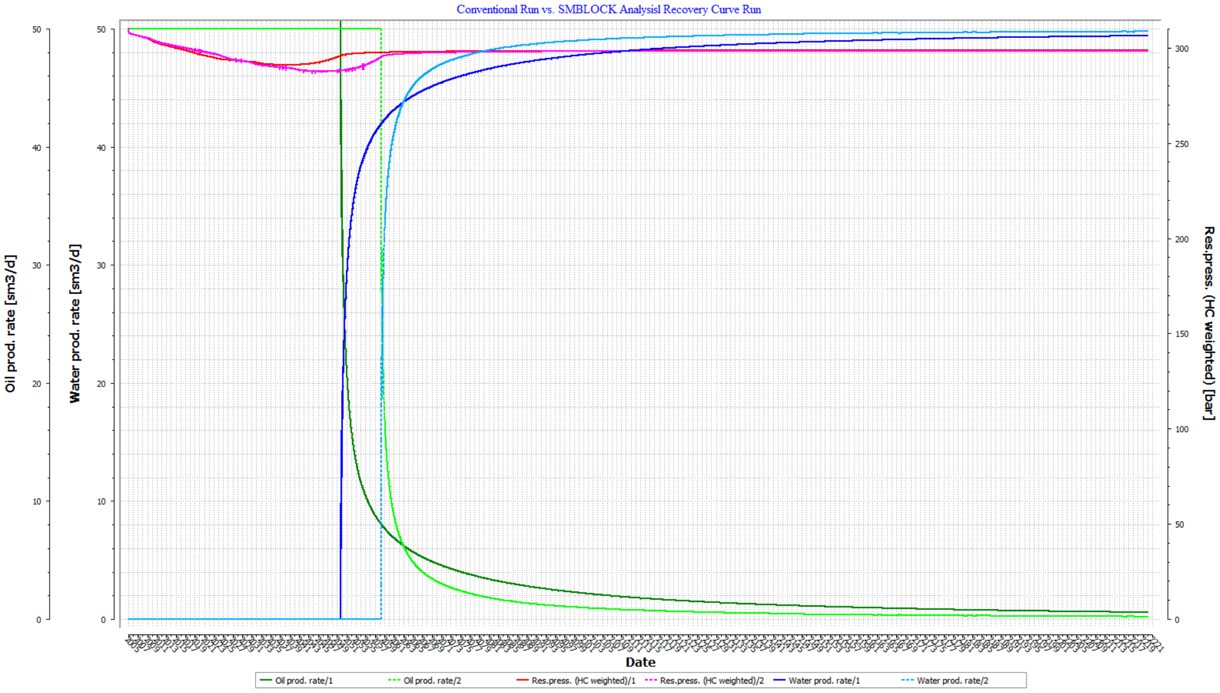


Figure 6.10: *Case IV*: Reservoir Production Rates and Pressure

6.6.1 Discussion

Figure 6.10 is the plot of water production rate (in dark blue), oil production rate (in dark green) and average reservoir pressure (in red) for the conventional run of *Case III* (as the conventional run cannot consider the additional valuable information about matrix block ID distribution in cells) and the plot of water production rate (in light blue), oil production rate (in light green) and average reservoir pressure (in pink) for the recovery curve run of *Case IV*. As it can be observed, the difference is even larger than in *Case III*.

It is worthwhile to mention that the results of a modeling approach has no direct connection to the reality. The simulation results must be checked against the real data and the simulation model that best matches the reality (by considering the physics of the system), is a better one to use for prediction. But as the recovery curve method takes much more physical details into account and needs much less homogenizations than the conventional run, it is logical to expect that the results from the recovery curve approach are very closer to reality than the results from the conventional run. Therefore, one can argue that in Figure 6.10, the results from the recovery curve run should be trusted instead of the conventional run.

However, it is still necessary to perform a history matching to fine-tune the region recovery curves to get the best results from the recovery curve run.

Chapter 7

Conclusions

7.1 Introduction

In this chapter, the main findings and conclusions from this work are presented in brief.

7.2 Conclusions

- Conventional transfer function is accurate for fluid expansion and solution gas drives.
- For displacement processes and in instances of gravity or capillary drive, the conventional transfer term cannot accurately represent the physics, due to ignoring the effects of saturation distribution when calculating the driving forces.
- The conventional transfer function cannot consider the effects of anisotropy on the matrix-fracture transfer which can be outstanding.
- In calculation of the matrix-fracture transfer (as well as in the single matrix block calculations), the fluid saturations should be used as the boundary condition and not the phase potentials.
- The shape factor is a physical parameter that represents the geometry of the matrix block and should not be regarded as a meaningless history-matching parameter.
- The shape factor can be measured from the outcrops of the fractured formations.
- Heinemann-Mittermeir shape factor formula is the most general form of the shape factor which considers anisotropy and can be used for matrix blocks of any shape.
- Single Matrix Block Analysis, which was developed a standard model and a preprocessing tool, should be performed to study the detailed physical behavior of the matrix blocks under different conditions and to generate the recovery curves for the matrix block under those conditions.
- In addition to using the SMB analysis, the recovery curves can be generated from the output of a matrix cell in a conventional run or measured in the laboratory.
- The derived dimensionless time can be used to classify many matrix blocks of different properties and shapes and describe them with one scaled recovery curve.
- The distribution of matrix blocks of different classes in the reservoir defines recovery curve regions with a lumped recovery curve which combines the effects of all the matrix blocks in that recovery curve region under different conditions.
- Recovery curve method is a more accurate yet practical approach to simulate NFRs.

7.3 Future research and development ambitions

- Developing the Single Matrix Block (SMB) Analysis also for compositional fluid description.
- Developing the possibility to study hysteresis in SMB Analysis.
- Studying the effect of hysteresis and oil resaturation on recovery curves.
- Applying recovery curve method for compositional fluid description.
- Investigation of matching the recovery curves based on historical data.

Chapter 8

References

[1] Abrahem, F.A.S.: “A New Approach to History Matching of Water Driven Oil Reservoirs”, PhD Thesis, Montanuniversität Leoben, Oct.2009.

[2] Anton, H., Rorres, C.: “Elementary Linear Algebra (Applications Version) 8th Edition”, John Wiley & Sons, ISBN 978-0-471-17052-5, 2000.

[3] Aguilera, R.: “Naturally Fractured Reservoirs”, Penn Well Books, Tulsa, Oklahoma, 1980, p.8.

[4] Amiry, M.T.: “Importance and Applicability of a Generalized Shape Factor by Modeling Dual Porosity Reservoirs”, ECMOR XIII Conference Proceedings, 2012. http://earthdoc.eage.org/publication/publicationdetails/?publication=62242

[5] Asadi, M., Ghalambor, A., Rose, W.D., Shirazi, M.K.: “Anisotropic Permeability Measurement of Porous Media: A 3-Dimensional Method”, SPE Conference Paper, SPE 59396, 2000.

[6] Baecher, G.B., Lanney, N.A., and Einstein, H.H.: “Statistical Description of Rock Properties and Sampling”, presented at the 18th US Symposium on Rock Mechanics, 1987.

[7] Baker, R.O., Schechter, D.S., McDonald, P., Knight, W. and Leonard, P.: “Development of a Fracture Model for Spraberry Field, Texas USA”, SPE 71635, 2001.

[8] Barenblatt, G. J., Zheltov, I. P., and Kochine, I. N.: “Basic Concepts in the Theory of Seepage of Homogeneous Liquids in Fissured Rocks”, J. Appl. Math. Mech., 1286-1303, p.24, 1960.

[9] Barker, J.A.: “Block-Geometry Functions Characterizing Transport in Densely Fissured Media”, Journal of Hydrology, Vol. 77, 1985, pp.263-279.

[10] Beckner, B.L.: “Improved Modeling of Imbibition Matrix/Fracture Fluid Transfer in Double Porosity Simulators”, PhD Dissertation at Stanford University, 1990.

[11] Blair, P.M.: “Calculation of oil displacement by intercurrent water imbibition”, paper no. 1475G, *Trans. AIME* secondary Rec. Symp., Wichita Falls, Texas, U.S.A., May 1960.

[12] Bloomfield, J.: “Characterization of Hydrogeologically Significant Fracture Distributions in the Chalk: an Example from the Upper Chalk of Southern England”, Journal of hydrology, 184, 335-379, 1996.

[13] Bourbiaux, B.J., Cacas, M.C., Sarda, S., and Sabathier, J.C.: “A Fast and Efficient Methodolody to Convert Fractured reservoir Images Into a Dual-Porosity Model”, SPE 38907, 1997.

[14] Bourbiaux, B., Basquet, R., Cacas, M., Daniel, J. and Sarda, S.: “An Integrated Workflow to Account for Multi-Scale Fractures in Reservoir Simulation Models: Implementation and Benefits”, SPE 78489, 2002.

[15] Bratton, Tom et al.: “The Nature of Naturally Fractured Reservoirs”, Schlumberger Oilfield Review, Summer 2006.

[16] Buckley, S.E. and Leverett, M.C.: “Mechanism of Fluid Displacement in Sands”, Trans AIME, Vol. 146, 1942, pp.107-116.

[17] Cacas, M.C., Ledoux, E., de Marsily, G. and Tillie, B:. “Modelling Fracture Flow With a Stochastic Discrete Fracture Network: Calibration And Validation”, Water Resources Research, 26, pp. 479-489, 1990.

[18] Cardwell, W.T., Parsons, R.L., “Average permeabilities of heterogeneous oil sands”, Trans. Am. Inst. Mining. Met. Pet. Eng., 1945, pp.34-42.

[19] Chaobal, A.N., Rangel-German, E.R. and Kovscek, A.R.: “Experimental Determination of Time-Dependent Matrix-Fracture Shape Factors for Different Geometries and Fracture Filling Regimes”, paper SPE 92131, 2004.

[20] Chen, J., Miller, M.A. and Sepehrnoori, K.: “Theoretical Investigation of Countercurrent Imbibition in Fractured Reservoir Matrix Blocks”, paper SPE 29141 presented at the SPE Symposium of Reservoir Simulation, San Antonio, TX, USA, 12-15 February, 1995.

[21] Coats, K.H.: “Implicit Compositional Simulation of Single-Porosity and Dual-Porosity Reservoirs”, Presented at the First International Forum on Reservoir Simulation, Alpbach, September 12-16, 1988 and SPE 18427.

[22] Creative Dimension Software Ltd - 3D Software Modeller Pro.

[23] Dagan, G., “Flow and Transport in Porous Formations”, Springer-Verlag, New York, 1989.

[24] da Silva, F.V., and Belery, P.: “Molecular Diffusion in Naturally Fractured Reservoirs: A Decisive Recovery Mechanism”, SPE 19672, presented at the 64th Annual Technical Conference and Exhibition, San Antonio, Texas, October 8-11, 1989.

[25] Deng, B., Lee, C., and Chang, J.: “Characterization and Interpretation of Variability of Fracture pattern in Jointed Andesites, Lan-Yu, Taiwan”, Journal of the Chinese Institute of Engineers, vol.18, No. 6, 787-799, 1991.

[26] Dershowitz, B., LaPointe, P. Eiben, T. and Wei, L.: “Integration of Discrete Feature Network Methods With Conventional Simulator Approaches”, SPE 62498, 1998.

[27] Dershowitz, W., Shuttle, D. and Parney, R.: “Improved Oil Sweep through Discrete Fracture Network Modeling of Gel Injections in the South Oregon Basin Field, Wyoming”, SPE 75162, 2002.

[28] de Swaan, A.: “Analytic Solutions for Determining Naturally Fractured Reservoir Properties by Well Testing”, SPE 5346-PA, 1976.

[29] de Swaan, A.: “Influence of Shape and Skin of Matrix-Rock Blocks on Pressure Transients in Fractured Reservoirs”, SPE Formation Evaluation, pp. 344-352, Dec. 1990.

[30] Dunham, W.: “Heron’s Formula for Triangular Area, Ch. 5 in Journey of the Genius: The Great Theorems of Mathematics.”, New York: Wiley, pp. 113-132, 1990.

[31] Durlofsky, L.J.: “Numerical Calculation of Equivalent Grid Block Permeability Tensors for Heterogeneous Porous Media”, Wat. Resou. Res., 27(5), 699-708, 1991.

[32] Engelder, T.: “Loading Paths to Joint Propagation During a Tectonic Cycle: An Example from the Appalachian Plateau, USA”, J. Struct. Geol., Vol. 7, 1987, pp. 450-476.

[33] Famy, C., Bourbiaux, B. and Quintard, M.: “Accurate Modeling of Matrix-Fracture Transfer in Dual Porosity Models: Optimal Subgridding of Matrix Blocks”, paper SPE 93115 presented at the 2005 SPE Reservoir Simulation Symposium, Houston, TX, USA, 31 January-2 February 2005.

[34] Gale, J.E., Schaefer, R.A., Carpenter, A.B., and Herbert, A.: “Collection, Analysis, and Integration of Discrete Fracture Data from the Monterey Formation for Fractured Reservoir Simulation”, paper SPE 22741 presented at the SPE Annual Technical Conference and Exhibition, Dallas, USA, October 1991.

[35] Gerke, H.H. and van Genuchten, M.Th.: “Macroscopic Representation of Structural Geometry for Simulating Water and Solute Movement in Dual-Porosity Media”, Advances in Water Resources, Vol.19, No.6, 1996., pp.343-357.

[36] Gilman, J.R., and Kazemi, H.: “Improved Calculation for Viscous Gravity Displacement in Matrix Blocks in Dual-Porosity Simulators”, SPE JPT, January, 60-70, 1988 and SPE 16010.

[37] Gilman, J.R.: “Practical Aspects of Simulation of Fractured Reservoirs,” paper presented at the Seventh International Forum on Reservoir Simulation, Baden-Baden, Germany, 2003.

[38] Graham, J.W. and Richardson, J.G: Note not published, Referenced in Collins,R.E.: “Flow of Fluids through porous materials”, Reinhold Publishing Corporation, New York, 1961, p.166.

[39] Gross, M.R., and Engelder, T.: “Strain Accomodated by Brittle Failure in Adjacent Units of the Monterey Formation, U.S.A.: Scale Effects and Evidence for Uniform Displacement Boundary Conditions”, Journal of Structural Geology, Vol. 17, No. 9, 1303-1318, 1995.

[40] Gudmundsson, A.: “Tectonics of the Thingvellir Fissure Swarm, SW Iceland”, Journal of Structural Geology, vol 9, No. 1, 61-69, 1970.

[41] Gurpinar, O. and Kossack, C.A.: “Realistic Numerical Models for Fractured Reservoirs”, paper SPE 59041 presented at the SPE International Petroleum Conference and Exhibition, Mexico, 1-3 February 2000.

[42] Heinemann, Z.E.: Personal Communications, 2009-2013.

[43] Heinemann, Z.E.: “Introduction to Simulation of Dual Porosity Reservoirs”, Textbook Montanuniversität Leoben, Austria, 2005.

[44] Heinemann, Z.E.: “Using Recovery Curves in Modeling Natural Fractured Hydrocarbon Reservoirs”, Proposal for a PhD Research Project at the Montanuniversität Leoben, Austria, March 2004.

[45] Heinemann Oil GmBH: “Pars Reservoir Simulator, Version 2.1”, June 2012.

[46] Heinemann Oil GmbH: “***PRS*** Time-Dependent Data Manual, Version 2.1”, June 2012.

[47] Heinemann Oil GmbH: “***PRS*** Technical Description, Version 2.1”, June 2012.

[48] Heinemann, Z.E., and Mittermeir, G.M.: “Derivation of the Kazemi-Gilman-Elsharkawy, Generalized Dual Porosity Shape Factor”, Transp. Porous Media, Sep. 2011.

[49] http://en.wikipedia.org/wiki/Permkeability_(earth_sciences)

[50] http://nendowingsmirai.yuku.com/topic/6964/Voronoi-fracture-or-3D-Voronoi-How?page=2

[51] Jamison, R.W.: “Quantitative Evaluation of Fractures on Mookshood Anticline a Detachment Fold in the Foothills of Western Canada”, AAPG Bulletin, vol. 81, No. 7, pp. 1110-1132, 1997.

[52] Kazemi, H., Gilman,J.R. and El-Sharkawy, A.M.: “Analytical and Numerical Solution of Oil Recovery from Fractured Reservoirs Using Empirical Transfer Functions”, SPE Reservoir Engineering, May 1992, pp.219-227.

[53] Kazemi, H., Merril, L.S., Porterfield, K.L., and Zeman, P.R.: “Numerical Simulation of Water-Oil Flow in Naturally Fractured Reservoirs”, SPEJ December, pp.317-326 and SPE Paper 5719, 1976.

[54] Kossack, C.A.: “Simulation of Gas/Oil Displacements in Vuggy and Fractured Reservoirs”, paper SPE 101674 presented at the 2006 SPE Annual Technical Conference and Exhibition, San Antonio, TX, USA, 24-27 September 2006.

[55] Lash, Gary G.: “The Upper Devonian Rhinestreet Shale: An Unconventional Reservoir in Western New York State”, Oral presentation at AAPG Annual Convention, Search and Discovery Article #10108, Houston, Texas, April 9-12, 2006.

[56] Lee, S.H., Jensen, C.L. and Lough, M.F.: “An Efficient Finite Difference Model For Flow in a reservoir with Multiple Length-Scale Fractures”, SPE 56752, 1999.

[57] Lewandowska, J., Szymkiewicz, A., Burzynski, K. and Vauclin, M.: “Modeling of Unsaturated Water Flow in Double-Porosity Soil by the Homogenization Approach”, Advances in Water Resources, Vol.27, 2004, pp.283-296.

[58] Lim, K.T, Aziz, K.: “Matrix-fracture transfer shape factors for dual-porosity simulators”, Journal of Petroleum Science and Engineering 13, 1995.

[59] Lishman, J.R.: “Core Permeability Anisotropy”, Journal of Canadian Petroleum Technology, 9(2), 1970.

[60] Matherson, G., “Elements pour une Thèorie des Milieux Poreux”, Masson, Paris, 1967.

[61] Mathias, S.A. and Zimmerman, R.W.: “Laplace transform inversion for late-time behavior of groundwater flow problems”, Water Resources Research, 39(10), p.1283, 2003.

[62] Mattax, C.C. and Kyte, J.R.: “Imbibition Oil Recovery from Fractured, Water-Drive Reservoirs”, SPE Journal 177-84, June 1962, Trans. AIME, 225.

[63] Miller, N.: “Predicting Flow Characteristics of a Lixiviant in a Fractured Crystalline Rock Mass”, Report of Investigations 9457, Bureau of Mines (1993), 24.

[64] Moustov, A., Pervago, E., Shevnin, V.: “A New Approach to Resistivity Anisotropy Measurements”, SEG 2000 Expanded Abstracts, 1381, 2000.

[65] Muskat, M.: “Physical Principles of Oil Production”, McGraw-Hill Book Company, New York, 1949.

[66] Muskat, M. and Taylor, M.O.: “Effect of Reservoir Fluid and Rock Characteristics on Production Histories of Gas-Drive Reservoirs”, Trans., AIME 1946, 165, pp.78-93.

[67] Muskat, M.: “The Flow of Homogeneous Fluids Through Porous Media”, 1937, reprinted 1982, IHRDC, Boston, Mass., pp.225-227.

[68] Narr, W., Schechter, D.W. and Thompson, L.B.: “Naturally Fractured Reservoir Characterization”, ISBN 978-55563-112-3 Society of Petroleum Engineers, Richardson, TX, USA, 2006.

[69] Oda, M., Hatsuyama, Y. and Ohnishi, Y.: “Numerical Experiments on Permeabilty Tensor and its application to jointed granite at Stripa Mine, Sweden”, Journal of Geophysical Research, 92, pp. 8037-8048, 1987.

[70] Onsager, L.: “Reciprocal Relations in Irreversible Processes”, Phys. Rev. 37, 405-426, 1931.

[71] OpenFlowSuite, “The IFP-Group geoscience software solution for reservoir assessment & basin modeling”, http://www.openflowsuite.com/.

[72] Ouenes, A., Feng Shen: “Fracture Characterisation”, paper presented at the Sevent International Forum on Reservoir Simulation, Baden-Baden, 2003

[73] Ouenes,A., Hartley, L.J.: “Integrated Fracture Modeling Using Both Discrete and Continuum Approaches”, SPE 62939 presented at the 2000 SPE Annual Technical Conference and Exhibition in Dallas, Texas, 1-4 Oct.2000.

[74] Panek, L.A.: “Estimating Fracture Trace Length form Censored Measurements on Multiple Scanlines”, presented at the 1985 Proceedings of the International Symposium on Fundamentals of Rock Joints, Björkliden, 15 - 20 September 1985.

[75] Penuela, G., Civan, F., Hughes, R. G. and Wiggins, M.L.: “Time-Dependent Shape Factor for Interporosity Flow in Naturally Fractured Gas-Condensate Reservoirs”, paper SPE 75524 presented at the SPE Gas Technology Symposium held in Calgary, Alberta, Canada, 30 Apr. - 2 May 2002.

[76] Penuela, G., Hughes, R. G., Civan, F. and Wiggins, M.L.: “Time-Dependent Shape Factors for Secondary Recovery in Naturally Fractured Reservoirs”, paper SPE 75234, April 2002.

[77] Pfeiffer, P.: “Characterization of Fractured Carbonate Reservoirs”, Master’s Thesis, Montanunivetsität Leoben, June 2006.

[78] Pirker, B.,: “Investigations of Recovery Mechanism for Fractured Dual-Porosity Reservoirs with Bottom-Water Drive”, Diploma Thesis, Montanuniversität Leoben, 2005.

[79] Pirker, B., Heinemann, Z.E.: “Method to Preliminary Estimation of the Reserves and Production Forecast for Dual Porosity Fractured Reservoirs”, paper SPE 113378, prepared for presentation at the 2008 SPE EUROPEC/EAGE Annual Conference and Exhibition held in Rome, Italy, 9 - 12 June 2008.

[80] Pirker B., Harrer, A.W., Heinemann, Z.E.: “Reserve Estimation for Naturally Fractured Reservoirs Using Numerically Derived Recovery Curves”, prepared for presentation at the 11th European Conference on the Mathematics of Oil Recovery (ECMOR XI) held in Bergen, Norway, 8 - 11 September 2008.

[81] Pirker, B.: “A New Approach for Modeling Dual Porosity Reservoirs Using Recovery Curves”, PhD Dissertation, Montanuniversität Leoben, Nov.2008.

[82] Pirson,S.J.: “Oil Reservoir Engineering”, McGraw-Hill Book Company, New York, 1958.

[83] Pruess, K. and Narasimhan, T.N.: “A Practical Method for Modeling Fluid and Heat Flow in Fractured Porous Medium”, SPEJ February, 14-26 and SPE Paper 10509, 1985.

[84] Quandalle, P. and Sabathier, J.C.: “Typical Features of a Multipurpose Reservoir Simulator”, SPE Reservoir Engineering November 1989.

[85] Quintard, M. and Whitaker, S.: “Theoretical development of region-averaged equations for slightly compressible single-phase flow”, Adv. Water Res., 19(1), 29-47, 1996.

[86] Rangel-German E.R. and Kovscek A.R.: “Time-Dependent Matrix-Fracture Shape Factors for Partially and Completely Immersed Fractures”, paper SPE 84411 presented at the SPE ATCE held in Denver, Colorado, USA, 5-8 Oct. 2003.

[87] Rives, T., Razack, M., Petit, J.P., and Rawnsley, K.D.: “Joint Spacing: Analogue and Numerical Simulations”, Journal of Structural Geology, Vol. 14, No. 8/9, pp. 925-937, 1992.

[88] Rose, W.: “A New Method to Measure Directional Permeability”, Journal of Petroleum Technology, May 1982.

[89] Rosen, R,H. and Shen,E.L.: “Simulation of Gas/Oil Drainage and Water/Oil Imbibition in Naturally-Fractured Reservoirs”, paper SPE 16982 presented at the 1987 SPE Annual Technical Conference and Exhibition, Dallas, TX, Sept. 27-30,1987.

[90] Rouleau, A. and Gale, J.E.: “Statistical Characterization of the Fracture System in the Stripa Granite, Sweden”, International Journal of Rock Mechanics and Mineral Sciences & Geomechanical Abstracts (1985) 22, No. 6, 353-367

[91] Roxar training, “Naturally Fractured Reservoirs: An Introduction to their Appraisal and Management”, support.roxar.com.

[92] Sarda, S., Jeannin, L., Basquet, R. and Bourbiaux, B.: “Hydraulic Characterization of Fractured Reservoirs: Simulation on Discrete Fracture Models”, SPE 77300, 2001.

[93] Sarma, P.: “New Transfer Functions For Simulation of Naturally Fractured Reservoirs with Dual Porosity Models”, Master Thesis, Petroleum Dept. Stanford University, 2003.

[94] Sarma, P., Aziz, K.: “Production Optimization with Adjoint Models Under Nonlinear Control-State Path Inequality Constraints”, SPE Conference Paper, 99959-MS. 2006.

[95] Schlumberger: “ECLIPSE Technical Description, 2004A_1”, 1982-2004.

[96] Thomas, L.K., Dixon, T.N., and Pierson, R.G.: “Fractured Reservoir Simulation”, SPE Journal, 638-648 (Feb. 1983) Trans. AIME, 243.

[97] Ueda, Y., Murata, S., Watanabe, Y. and Funatsu, K.: “Investigation of the Shape Factor Used in the Dual-Porosity Reservoir Simulator”, paper SPE 19469 presented at the SPE Asia-Pacific Conference held in Sydney, Australia, 13-15 Sept. 1989.

[98] Walter D., R.: “A New Method to Measure Directional Permeability”, Journal of Petroleum Engineering, 34(5), 1142-1144, 1982.

[99] Warren, J.E., and Root P.J.: “The Behavior of Naturally Fractured Reservoirs”, SPEJ, September, 245-255, 1963, Trans.AIME, 228.

[100] Weitzenböck, J.R., Shenoi, R.A., Wilson, P.A.: “Measurement of three-dimensional permeability”, Applied Science and Manufacturing, 29, 159-169, 1997.

[101] Wolfram MathWorld, http://mathworld.wolfram.com/Point-PlaneDistance.html

[102] www.netl.doe.gov/technologies/oil-gas/NaturalGas/Projects_n/EP/DCS/images/lpf31063FracturedCore.jpg (August, 2008)

[103] Zellou, A.M.: “Seismically Driven Improved Fractured Reservoir Characterization”, paper SPE 92031, SPE International Petroleum Conference, Mexico, 2004.

[104] Zimmerman, R.W., Chen, G., Hadgu, T. and Bodvarsson, G.S.: “A numerical dual-porosity model with semi-analytical treatment of fracture/matrix flow”, Water Resources Research, 29(7), pp. 2127-2137, 1993.

Appendix A

The Mathematical Model

A.1 Introduction

In this chapter, mathematical modeling of naturally fractured reservoirs is briefly explained.

While mathematically modeling fractured oil and gas reservoirs, two fundamentally different approaches are possible:

The first approach commonly used to model fractured reservoirs is the *Discrete Fracture Network* (*DFN*) where the reservoir volume is filled with fractures represented by planes or disks. In the *DFN* model, each conductive fracture is modeled explicitly as one or more 1D, 2D, or 3D element. Physical properties such as transmissivity or storage, and geometrical properties such as size, elongation and orientation are assigned to each polygon based upon measured data or geologically conditioned statistical distributions derived from measured values. However, gathering the additional data is rather difficult and more importantly, the required calculation cost (in terms of CPU and memory) for large carbonate fields (such as the ones existing in the middle east) makes this method currently industrially impractical. The difficulty to model fluid flow in such a model is obvious. Applying this method to the current well-known commercial industrial-scale reservoir simulators is not possible and is not tried.Theoretical models can be created but it is difficult to imagine how they could be coupled with the geologic realism.

The second approach uses a *Continuous Framework* model (*CFM*) where many geologic drivers could be incorporated in creating an integrated fracture model. These continuous fracture models are described with properties as fracture intensity (fracture count), orientation, connectivity, etc. offering the possibility to derive the usual rock properties as porosity, directional permeability and relationship to the rock matrix between the fractures. The approach leads to the *Dual-Continuum* concept. In every point of the space two values of each properties exist, one for the fracture the other for the matrix continuum.

In the presented research project the second approach will be applied only. The actual chapter is extracted from the work of Prof. Heinemann's Lecture Notes "*Introduction to Simulation of Dual-Porosity Reservoirs*"[43] and other underlaying publications of the author.

A.2 Dual-Continuum Model

It will be assumed that two continua exist in the same space, both fill the space completely without gaps and overlaps. One continuum represents the fractures and one the matrix. Both continua have the same type of properties and their own variables. The matrix properties and variables are signed by the subscript *m* and the fracture by *f*. For example the porosities are ϕ_m, and ϕ_f which are functions of the pressure p_m and p_f, respectively. The pore space is occupied by fluids and separated by phase surfaces. The proportion of one phase in the pore volumes are defined as the phase saturations S_{pm} and S_{pf}.

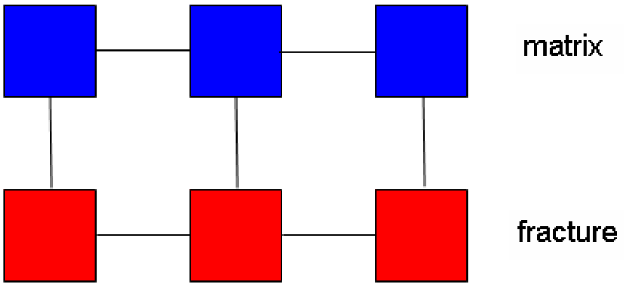


Figure A.1: Dual-porosity dual-permeability concept

The balance equation for the fracture continuum:

$$-\nabla\left(\sum_{p=1}^{P}\vec{u}_p D_p x_{pc} M_c\right)_f - \nabla\left(\sum_{p=1}^{P}\overrightarrow{J_{pc}} M_c\right)_f + q_{cf}M_c - q_{cmf}M_c \\ = \frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P} S_p D_p x_{pc} M_c\right)_f \tag{A.1}$$

The balance equation for the matrix continuum:

$$-\nabla\left(\sum_{p=1}^{P}\vec{u}_p D_p x_{pc} M_c\right)_m - \nabla\left(\sum_{p=1}^{P}\overrightarrow{J_{pc}} M_c\right)_m + q_{cm}M_c + q_{cmf}M_c \\ = \frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P} S_p D_p x_{pc} M_c\right)_m \tag{A.2}$$

Note that the third term in the left side of the Equation A.3 is split into two terms. The source/sink term q_{cm} and q_{cf} represent is the wells, while q_{cmf} is the matrix-fracture transfer term. The Equation A.1 and Equation A.2 can be rewritten in terms of mole balance rather than mass balance by using Equation A.17 and Equation A.24 in Equation A.1, to get the fracture mole balance equation,

$$\nabla\left(\sum_{p=1}^{P}\Lambda_{pc}k_a\bar{\bar{k}}'\nabla\Phi_p\right)_f + \nabla\left(\sum_{p=1}^{P}\phi S_pD_p\mathrm{D}_{pc}\nabla x_{pc}\right)_f + q_{cf} - q_{cmf} = \frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P}S_pD_px_{pc}\right)_f \tag{A.3}$$

and using Equation A.17 and Equation A.24 in Equation A.2, to gives the matrix mole balance equation.

$$\nabla\left(\sum_{p=1}^{P}\Lambda_{pc}k_a\bar{\bar{k}}'\nabla\Phi_p\right)_m + \nabla\left(\sum_{p=1}^{P}\phi S_pD_p\mathrm{D}_{pc}\nabla x_{pc}\right)_m + q_{cf} - q_{cmf} = \frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P}S_pD_px_{pc}\right)_m \tag{A.4}$$

Equation A.1 and Equation A.2 describe a dual-porosity dual-permeability case in which the fluids can flow within both domains as shown in Figure A.1.

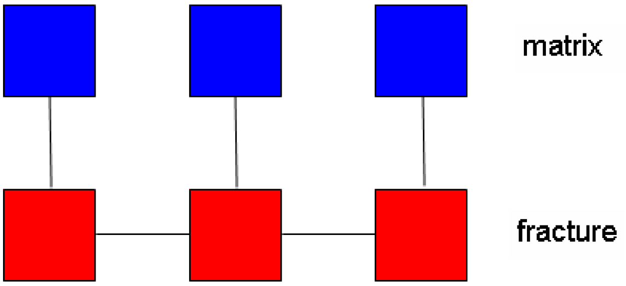


Figure A.2: Dual-porosity (single-permeability) concept

The porous material has a porosity ϕ, which is a function of the pressure (and temperature). The pore space is occupied by fluids and separated by phase surfaces. The proportion of one phase in the pore volume is defined as the phase saturation S_p. It is obvious that:

$$\sum_{p=1}^{P} S_p = 1. \tag{A.5}$$

Equation A.5 is called "*saturation constraint*". The fluid system is composed of *C* components. The composition of the p^{th} phase is given by the mole fractions x_{pc}; Equation A.6 is called "*mole constraint*".

$$\sum_{c=1}^{C} x_{pc} = 1; \qquad p \in \{1, \ldots, P\} \cdot \tag{A.6}$$

A.2.1 Convection and Diffusion Terms

Equation A.1 and Equation A.2 assume that two transport mechanisms have influence on the mass flow rate for which the driving forces and parameters has to be specified:

- Convection, described by Darcy’s law and
- Diffusion, defined by Fick’s law.

The total molar flux [kmole/s^1/m^2] for the component *c,* in general*,* is:

$$\vec{U}_c = \sum_{p=1}^{P} \vec{u}_p D_p x_{pc} + \sum_{p=1}^{P} \vec{J}_{pc} \,. \tag{A.7}$$

The filtration velocity of the phase *p*, $\vec{u}_p$, is given by the Darcy's law (for the multi-phase case):

$$\vec{u}_p = -\lambda_p \bar{\bar{k}} \nabla \Phi_p \,, \tag{A.8}$$

where the permeability tensor, $\bar{\bar{k}}$, in general, is a symmetric tensor (see Section 3.3.8).

$$\bar{\bar{k}} = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix}, \text{where } k_{ij} = k_{ji}. \tag{A.9}$$

$$\lambda_p = \frac{k_{rp}}{\mu_p} \text{, is the phase mobility, and} \tag{A.10}$$

$$\nabla \Phi_p = \nabla p_p - \rho_p \vec{g} \text{ , the potential gradient, and} \tag{A.11}$$

$$D_p = D_p(p_p, T_p, \bar{x}_{pc}) \text{, the specific molar density.} \tag{A.12}$$

The phase densities, ρ_p can be calculated as:

$$\rho_p = D_p \sum_{c=1}^{C} M_c x_{pc} \,. \tag{A.13}$$

The component mass mobility, Λ_{pc} , is:

$$\Lambda_{pc} = \lambda_p D_p x_{pc} \,. \tag{A.14}$$

To get an approximate value for Λ_p, some sort of averaging is necessary. For multi-phase flow upstream-weighting has shown to converge to the physically correct solution. The simplest way is to take one of the values, which leads to one-point upstream-weighting.[81] The differences between the phase pressures is the capillary pressure:

$$p_p - p_{p'} = P_{cpp'} \qquad p \neq p', p, p' \in \{1, \ldots, P\}. \tag{A.15}$$

J_{pc} [kmol.m^{-2}.s^{-1}] is the molar flux of the component c in phase p, which can be expressed by Fick's law:

$$\overrightarrow{J_{pc}} = -\phi S_p D_p \mathrm{D}_{pc} \nabla x_{pc}, \tag{A.16}$$

where D_{pc}[m²/s] is the diffusion coefficient. The diffusion coefficient is specific for the compound (solute) and depends on temperature and the composition of the system (solvent).

Very often the permeability of the matrix is of magnitudes less than the permeability of the fractures and/or is not continuous (i.e. it is broken into noncontinuous chunks). The matrix serves just as volume source or sink for the continuous fracture system, as shown in Figure A.2. Therefore, production is only possible from the fractures but not from the matrix. In this case Equation A.2 simplifies to:

$$q_{cmf} M_c = \frac{\partial}{\partial t}\left(\phi \sum_{p=1}^{P} S_p D_p x_{pc} M_c\right)_m, \tag{A.17}$$

where the matrix-fracture transfer q_{cmf} term includes convective and diffusive material transport as well. Figure A.3 shows a schematic of a dual-continuum model, including the transfer term.

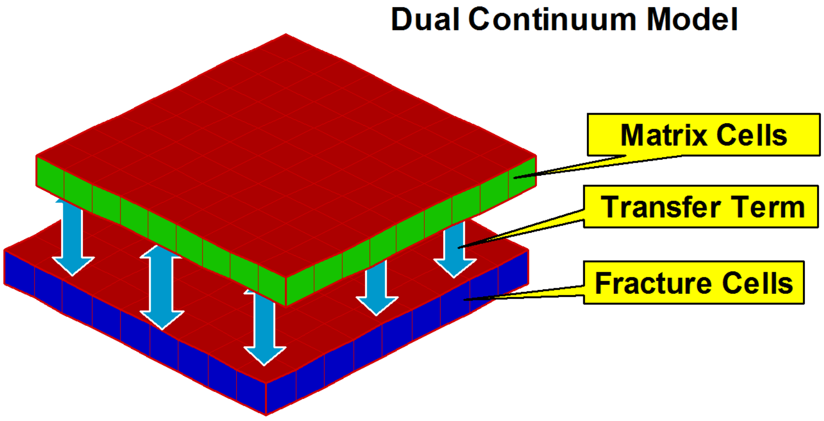


Figure A.3: Illustration of a dual-continuum model.

A.2.2 Application of Green Divergence Theorem

In order to be able to discretize the differential equations on a generalized grid, the control volume finite difference method (CVFD) method should be applied. This will be explained here just for one of the domains without using the subscript *m* or *f* but the same equations are valid for both continua. The Equation A.2 can be integrated over an arbitrary control volume V_i, leading to:

$$-\iiint_{V_i}\left[\nabla\left(\sum_{p=1}^{P}\vec{u}_p D_p x_{pc} M_c\right)\right]dV = \iiint_{V_i}\left[\frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P} S_p D_p x_{pc} M_c\right)\right]dV\,, \tag{A.18}$$

where the diffusion and production terms were disregarded for the sake of simplicity. V_i is a finite control volume around the grid point *i*.

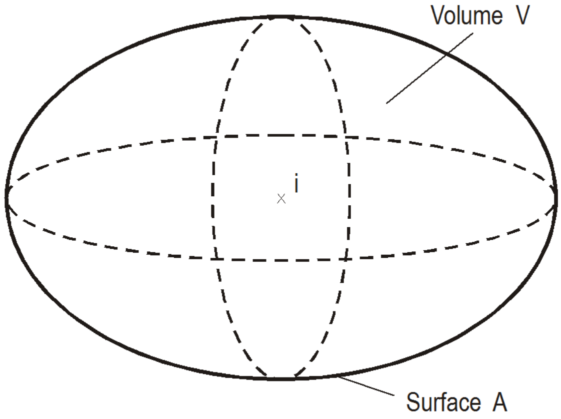


Figure A.4: Control volume around grid point

The flow term can be rewritten using the GREEN divergence theorem:

$$-\iiint_{V_i}\left[\nabla\left(\sum_{p=1}^{P}\vec{u}_p D_p x_{pc} M_c\right)\right]dV = -\iint_{A_i}\left(\sum_{p=1}^{P}\vec{u}_p D_p x_{pc} M_c\right)\vec{n}\,dA\ , \tag{A.19}$$

where $\iint_{A_i}$ denotes the integration over the total surface A_i of V_i and $\vec{n}$ is normal vector (the outward pointing unit vector) to the surface A_i. The same theorem can be applied to the diffusion term in the same manner. Inserting into Equation A.18 yields

$$-\iint_{A_i}\left(\sum_{p=1}^{P}\vec{u}_p D_p x_{pc} M_c\right)\vec{n}\,dA = \iiint_{V_i}\left[\frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P} S_p D_p x_{pc} M_c\right)\right]dV\ . \tag{A.20}$$

Equation A.20 is a mass balance. But it is obvious that mass and mole balances are equivalent when cancelling the constant mole mass M_c from both sides. This leads to a mole balance equation:

$$-\iint_{A_i}\left(\sum_{p=1}^{P}\vec{u}_p D_p x_{pc}\right)\vec{n}\,dA = \iiint_{V_i}\left[\frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P}S_p D_p x_{pc}\right)\right]dV \quad . \tag{A.21}$$

Substituting the velocity according to Darcy's law and splitting up the double integral $\iint_{A_i}$ into a sum over all surfaces A_{ij} connecting the block i with its neighbors j, yields for the flow term:

$$-\iint_{A_i}\left(\sum_{p=1}^{P}\vec{u}_p D_p x_{pc}\right)\vec{n}\,dA = \sum_{j=1}^{N}\iint_{A_{ij}}\left(\sum_{p=1}^{P}\Lambda_{pc}k_a\bar{\bar{k}}'\nabla\Phi_p\right)\vec{n}\,dA \ , \tag{A.22}$$

where already the component mobility and the normalized permeability tensor

$$\Lambda_{pc} = \lambda_p D_p x_{pc} \quad \text{and} \quad \bar{\bar{k}} = k_a\bar{\bar{k}}', \tag{A.23}$$

were substituted. A similar expression can be written for the diffusion term:

$$-\iint_{A_i}\left(\sum_{p=1}^{P}\overrightarrow{J_{pc}}M_c\right)\vec{n}\,dA = \sum_{j=1}^{N}\iint_{A_{ij}}(\phi S_p D_p D_{pc}\nabla x_{pc})\vec{n}\,dA \quad . \tag{A.24}$$

Now the fluid flow equation in integral formulation has the following form:

$$\sum_{j=1}^{N}\iint_{A_{ij}}\left(\sum_{p=1}^{P}\Lambda_{pc}k_a\bar{\bar{k}}'\nabla\Phi_p\right)\vec{n}\,dA + \sum_{j=1}^{N}\iint_{A_{ij}}(\phi S_p D_p D_{pc}\nabla x_{pc})\vec{n}\,dA + q_c \pm q_{cmf} = \iiint_{V_i}\left[\frac{\partial}{\partial t}\left(\phi\sum_{p=1}^{P}S_p D_p x_{pc}\right)\right]dV \tag{A.25}$$

where q_c is the discrete sink/source term for the component c acting in the volume V_i.

A.2.3 Matrix-fracture transfer term

The matrix-fracture fluid transfer while numerical modeling of dual-porosity reservoirs has ever been and is still an area of extensive research and discussion. Warren and Root[99] gave the first definition of the shape factor when they introduced their idealized sugar cube model of fractured porous rocks. One of the most accepted and widely implemented shape factor definitions used in numerical modeling was given by Kazemi *et al.*[52]:

$$\sigma = 4\left(\frac{1}{L_x^2} + \frac{1}{L_y^2} + \frac{1}{L_z^2}\right) . \tag{A.26}$$

Coats[21] recommended a shape factor that is twice the value of Kazemi *et al.*[52]. All mentioned authors assumed the mass transfer between matrix and fracture to be pseudo-steady state. Numerous other authors proposed shape factors based on numerical and/or laboratory experiments. Vicarious Thomas *et al.*[96] and Ueda *et al.*[97] are mentioned. Gilman[37] claims that the shape factor is a function of fracture spacing (or intensity), and is not inherently a time-dependent parameter.

To the knowledge of the author, Barker[9] presented the most general formulation of the matrix-fracture transport equations so far. In his formulation the matrix block has an arbitrary shape with a volume to surface ratio of *a*. The boundary value problem leads to a solution containing the time depending dimensionless Block-Geometry Function (*BGF*). The *BGF* depends on the geometry of the characteristic matrix block but not on its absolute size. The volume to area ratio of the blocks captures the block shape. Assuming pseudo- (quasi-)steady state conditions, the BGF becomes time independent and leads to a definition of the shape factor. The shape factor after Barker[9] is:

$$\sigma = \alpha\left(\frac{V_m}{A_m}\right)^{-2} , \tag{A.27}$$

where V_m is the volume, A_m is the surface area and α is a dimensionless parameter.

Consequently, the assumption of pseudo-steady flow between matrix and fracture leads always to a constant parameter, called “shape factor”. Otherwise, the time depending Block-Geometry Function must be considered and a *function* should not be named as *factor*. Nevertheless, some authors (Penuela[75], Rangel-German and Kovscek[86]) published such functions calling them time-dependent shape factors. The supposition that the exchange term for all possible physical situations will depend on the geometry of the discontinuous element (the matrix) of the model was postulated from the beginning, at first time by Warren and Root[99]. It should be also mentioned that their sugar cube model and the resulting shape factor is based already on discretization and homogenization. The weakness of the concept was that the shape factor was defined for specified geometries. In reality, the matrix blocks are neither cubes nor spheres; therefore, it was necessary to find a general applicable geometrical measure.

Based on water imbibition experiments of Mattax and Kyte[62], Kazemi *et al.*[52] introduced a generalized shape factor given by Equation A.28. It is easy to show that for idealized (cuboid) cases this equation becomes identical to the Warren and Root[99] expression given in Equation A.29. Despite of this asymptotic accordance and its widely acceptance, its general validity was shown at first by Heinemann and Mittermeir[48]. The derivation of Heinemann and Mittermeir will be presented in the Appendix B in details.

$$\sigma = \frac{1}{V_m} \sum_{j=1}^{N} \frac{A_j}{\left|\vec{d}_j\right|}. \tag{A.28}$$

$$\sigma = 4 \cdot \left(\frac{1}{L_x^2} + \frac{1}{L_y^2} + \frac{1}{L_z^2} \right). \tag{A.29}$$

Also Gerke and van Genuchten[35] tried to find a generalized geometrical factor and found that such a factor must contain the relation of the surface to the matrix volume as given in Equation A.27. Their geometry factor still contains, similar to Barker[9], an imprecisely defined dimensionless geometry-dependent coefficient. It should be emphasized that the dual-continuum concept is one but not the only method to describe flow in heterogenous porous media. On the other hand, the dual-continuum approach is still the most accepted and a very successful method to describe oil and gas recovery from naturally fractured dual-porosity reservoirs. Lewandowska *et al.*[57] claim to have found a more general approach. Their work contains interesting ideas but does not give any experimental evidence or demonstrates its practical applicability.

A.2.4 Nomenclature

The terms used in this chapter are defined as follows:

N	-	number of neighboring blocks
τ_{ij}	-	interblock transmissibility, [m^3]
x_{pc}	-	mole fraction of component *c* in phase *p*, [-]
λ_p	-	mobility of the phase *p*, [1/cp]
D_p	-	specific mole density, [kmol/m^3]
Φ_p	-	phase potential, [bar]
q_p	-	source/sink term for phase, [m^3/day]
Δt	-	time step, [day]
V_i	-	grid block volume, [m^3]
Φ	-	porosity, [-]
S_p	-	phase saturation, [-]
Δ_t	-	time difference operator, $\Delta_t\Gamma = \Gamma^{n+1} - \Gamma^n$
n	-	subscript, denoting the time step, [-]
n+1	-	subscript, new time step, [-]

Appendix B

Derivation of a Generalized Shape Factor

B.1 Introduction

This chapter describes the derivation of Heinemann and Mittermeir[48] generalized shape factor which is used in this study as the basis for calculations and recommended to be used industrially as well.

B.2 Mathematical Derivation

This description is based on the original work of Heinemann and Mittermeir[48] but reformulated for black oil fluid description, instead of the more general compositional one. For sake of simplicity the diffusive and production terms were omitted too. For more details refer to the original article[48]. It will be assumed that two continua exist in the same space, both fill the space without gaps. One continuum represents the fractures and the other the matrix. The matrix properties and variables are signed by the subscript *m* and the fracture by *f*. Two balance equations have to be written, once with the subscript *m* and once with the subscript *f*.

The balance equation of oil for the fracture and matrix continua are:

$$\nabla\left(\frac{k_{ro}}{\mu_o B_o} k_a \bar{\bar{k}}' \nabla\Phi_o\right)_f - q_{omf} = \frac{\partial}{\partial t}\left(\frac{\phi S_o}{B_o}\right)_f , \qquad \text{(B.1)}$$

$$\nabla\left(\frac{k_{ro}}{\mu_o B_o} k_a \bar{\bar{k}}' \nabla\Phi_o\right)_m + q_{omf} = \frac{\partial}{\partial t}\left(\frac{\phi S_o}{B_o}\right)_m , \qquad \text{(B.2)}$$

where the diffusion and production terms were disregarded for the sake of simplicity.

q_{omf} is the matrix-fracture transfer term. Equation B.1 and Equation B.2 describe a dual-porosity dual-permeability case in which the fluids can flow within both domains.

The permeability tensor $\bar{\bar{k'}}$ in the equations above is already split into a normalized tensorial part and an effective (or apparent) scalar permeability, k_a. Therefore, the permeability tensor can be rewritten as:

$$\bar{\bar{k}} = \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{12} & k_{22} & k_{23} \\ k_{13} & k_{23} & k_{33} \end{pmatrix} = k_a \bar{\bar{k'}} = k_a \begin{pmatrix} k'_{11} & k'_{12} & k'_{13} \\ k'_{12} & k'_{22} & k'_{23} \\ k'_{13} & k'_{23} & k'_{33} \end{pmatrix} . \tag{B.3}$$

Muskat[67] suggested to calculate the effective permeability as the geometrical mean of the values in the principle permeability directions (α,β,γ); which generally do not coincide with the coordinate axes x,y,z:

$$k_a = \sqrt[3]{k_\alpha k_\beta k_\gamma} . \tag{B.4}$$

Generally, the matrix permeability is of magnitudes less than the fracture permeability and is not continuous (i.e. it is broken into noncontinuous chunks). The matrix serves just as volume source or sink for the continuous fracture system. Therefore, flow is only possible in the fractures but not in the matrix domain. Consequently, the flow term disappears and the Equation B.2 simplifies to

$$q_{omf} = \frac{\partial}{\partial t}\left(\frac{\phi S_o}{B_o}\right)_m . \tag{B.5}$$

On the other hand Equation B.2 can be applied directly to an isolated matrix block with arbitrary shape, shown in Figure B.1. Equation B.2 can be integrated over the matrix block volume leading to:

$$\iiint_{V_{mi}} \nabla\left(\frac{k_{ro}}{\mu_o B_o} k_a \bar{\bar{k'}} \nabla \Phi_o\right)_m dV = \iiint_{V_{mi}} \left[\frac{\partial}{\partial t}\left(\frac{\phi S_o}{B_o}\right)\right] dV . \tag{B.6}$$

V_{mi} is a finite control volume around the point of gravity *i*. The flow term can be rewritten using the GREEN divergence theorem:

$$\sum_{j=1}^{N} \iint_{A_{ij}} \nabla\left(\frac{k_{ro}}{\mu_o B_o} k_a \bar{\bar{k'}} \nabla \Phi_o\right)_m \overrightarrow{n_{ij}} dA = \iiint_{V_{mi}} \left[\frac{\partial}{\partial t}\left(\frac{\phi S_o}{B_o}\right)\right] dV , \tag{B.7}$$

where the double integral was spilt up into a sum over all surfaces A_{ij}, displayed in Figure B.1.

The matrix block is surrounded by fractures represented by the points *j* facing the surfaces A_{ij}.

$\vec{n}_j$ is the normal vector (the outward pointing unit vector) to the surface A_{ij}.

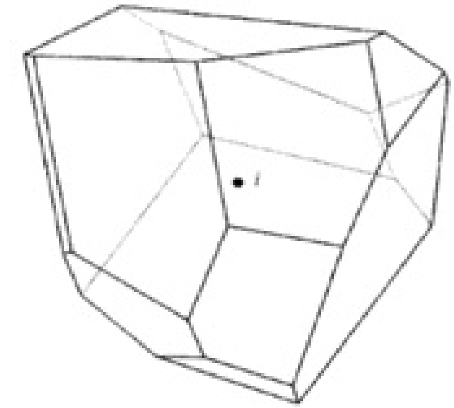


Figure B.1: General 3D grid matrix block

The dual-continuum approach handles the matrix-fracture transfer as a discrete term, expressed as the rate of a volume source. This rate can be derived from Equation B.7 by discretizing it in time and in space. The matrix block becomes then the control volume for which the flow term is integrated. Replacing the right side of Equation B.7 by average values:

$$\iint\int_{V_{mi}} \frac{\partial}{\partial t}\left(\frac{\phi S_o}{B_o}\right) dV = \frac{V_{mi}}{\Delta t}\Delta_t\left(\frac{\phi S_o}{B_o}\right)_{mi} , \tag{B.8}$$

results in:

$$\frac{1}{V_{mi}}\sum_{j=1}^{N}\iint_{A_{ij}} \nabla\left(\frac{k_{ro}}{\mu_o B_o} k_a \overline{\overline{k'}} \nabla\Phi_o\right)_m \vec{n}_{ij} dA = \frac{1}{\Delta t}\Delta_t\left(\frac{\phi S_o}{B_o}\right)_{mi} . \tag{B.9}$$

Equation B.9 is already divided by $V_{mi.}$. The permeability in Equation B.9 is the effective matrix-fracture permeability which, if the fracture block volume compared to the matrix block volume, is equivalent to the matrix permeability. Δt is the time step and Δ_t the time difference operator $\Delta_t x = x^{n+1} - x^n$. It is evident that the left side of Equation B.9 is the specific matrix-fracture flow rate of Equation B.5, therefore

$$q_{omf} = \frac{1}{V_{mi}}\sum_{j=1}^{N}\iint_{A_{ij}} \nabla\left(\frac{k_{ro}}{\mu_o B_o} k_a \overline{\overline{k'}} \nabla\Phi_o\right)_m \vec{n}_{ij} dA . \tag{B.10}$$

In order to be able to introduce this value in Equation B.1 and Equation B.2 the right hand side of Equation B.10 has to be discretized. This means that the surface integral will be replaced by multiplication with the area A_{ij}, the mobility will be determined from the average of the block state variables and the potential gradient by its component effective on the block sides.

Therefore, one has to approximate:

- the surface A_{ij} facing the matrix block *i* to the fracture side *j*,
- the oil mobility, $k_{ro}/(\mu_o B_o)$,
- the scalar product, $(\overline{\overline{k}}'\nabla\Phi_o)\overrightarrow{n_{ij}} = (\overline{\overline{k}}'\overrightarrow{n_{ij}})\nabla\Phi_o$.

The surfaces A_{ij} and the orientation of the unit normal vector $\vec{n}_{ij}$ are determined by the shape of the matrix block. Once the matrix block is identified, as in Figure B.1, the area and the orientation can be computed by means of geometry.

The mobility $k_{ro}/(\mu_o B_o)$, is required on the surface A_{ij}. To get an approximate value, some type of averaging is necessary. For multi-phase flow upstream-weighting has shown to converge to the physically correct solution. The simplest way is to take one of the values, which leads to one-point upstream-weighting.

For the approximation of the scalar product $(\overline{\overline{k}}'\nabla\Phi_p)\overrightarrow{n_j}$ imaginary grid points must be set into the surrounding fracture network to every block faces $j = 1, N$. The fracture width is assumed negligeable compared to the matrix block volume, therefore they are situated within the plane of the block sides. The fracture cells, containing a greater number of matrix blocks according the Warren-Root[99] concept is represented by the average pressure and saturations. Consequently, all of the neighboring fracture grid points have the same potential and can be set anywhere into the matrix block sides. The vector $\vec{d}_{ij}$ measuring the distance must be k-orthogonal to the surface A_{ij} between the point *i* and its neighbor *j*; therefore:

$$\overline{\overline{k}}'\overrightarrow{n_{ij}} \parallel \vec{d}_{ij} \quad ; \tag{B.11}$$

and so,

$$(\nabla\Phi_p \cdot (\overline{\overline{k}}'\vec{n}))_{ij} = \left|\overline{\overline{k}}'\overrightarrow{n_{ij}}\right| \frac{(\Phi_{oj} - \Phi_{oi})}{\left|\overrightarrow{d_{ij}}\right|} \quad . \tag{B.12}$$

Inserting Equation B.12 into Equation B.10 yields:

$$q_{omf} = \frac{1}{V_{mi}} \sum_{j=1}^{N} A_{ij} \frac{\left|\overline{\overline{k}}'\overrightarrow{n_{ij}}\right|}{\left|\overrightarrow{d_{ij}}\right|} \frac{k_{ro}}{\mu_o B_o} k_a(\Phi_{oj} - \Phi_{oi}) \quad . \tag{B.13}$$

The first of term of Equation B.13 contains only quantities depending on the geometry of the matrix block and its orientation in the coordinate system built by the principal permeability direction. These terms can be united into one constant, called, according the Warren-Root[99] concept, shape factor:

$$\sigma = \frac{1}{V_{mi}} \sum_{j=1}^{N} A_{ij} \frac{\left|\overline{\overline{k'}}\,\overrightarrow{n_{ij}}\right|}{\left|\overrightarrow{d_{ij}}\right|} , \tag{B.14}$$

which has to be seen as a characteristic parameter of the dual-continuum theorem and has a dimension L^{-2}. Shape factor as expected semantically, has to be a factor representing the effects of the shape of the system in the matrix-fracture transfer function and obviously it is expected that as long as the shape of the system (matrix shape, orientation etc.) has not changed, it remains constant. The resulting matrix-fracture transfer rate for a component is then

$$q_{omf} = \sigma \frac{k_{ro}}{\mu_o B_o} k_a (\Phi_{oj} - \Phi_{oi}) \ . \tag{B.15}$$

B.3 Nomenclature

The terms used in this chapter are defined as follows:

A	-	surface, [m^2]
B_o	-	formation factor, [-]
d_{ij}	-	distance, [m]
k_a	-	average permeability, [m^2]
k_{ro}	-	relative permeability of oil phase, [-]
N	-	number of neighboring blocks, [-]
$\vec{n}_{ij}$	-	normal unit vector of surface A_{ij}, [-]
q_{omf}	-	fracture-matrix transfer rate, [sm3/s]
S_o	-	phase saturation, [-]
p	-	pressure, [Pa]
t	-	time, [s]
Δt	-	time step, [s]
V_i	-	grid block volume, [m^3]
σ	-	shape factor, [m^{-2}]
μ	-	viscosity, [Pas]
Φ_o	-	phase potential, [Pa]
ϕ	-	porosity, [-]

Subscripts

o	-	oil
x,y,z	-	coordinate directions

Operators

Δ_t	-	time difference operator

Appendix C

Laboratory Experiment of Measuring Recovery Curve

C.1 Introduction

The recovery curve of any piece of rock can be measured in the laboratory with a method similar to what is explained in this chapter.

C.2 Water Imbibition Recovery Curve Experiment

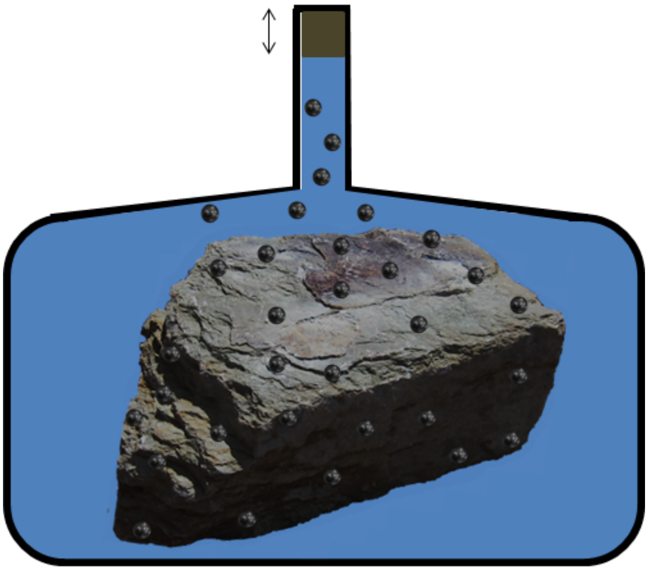


Figure C.1: Schematic representation of the laboratory apparatus for measuring the recovery of the matrix block submerged in water.

The experiment is performed in the following steps:

- The matrix block is saturated with oil. It is usually fully oil-saturated but for other studies it can also be initially saturated with any percentage with oil/water (and even gas) to study different conditions. The initial oil volume should be also measured before saturating the block to know the original oil in place for the matrix block.
- The matrix block is submerged in a pressure container full of water with the desired average pressure as shown in Figure C.1.
- The recovered oil volume is measured periodically.
- The plot of the recovery factor (= the recovered oil volume / the initial oil volume) versus time to create the recovery curve for each case.

C.3 Gas Gravity Drainage Recovery Curve Experiment

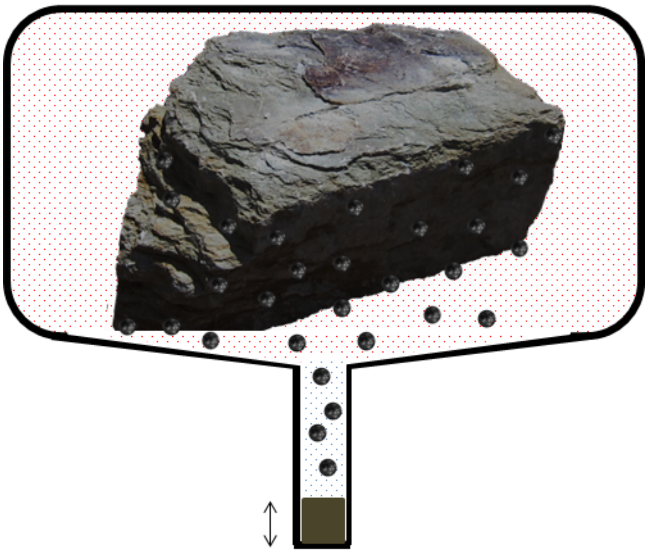


Figure C.2: Schematic representation of the laboratory apparatus for measuring the recovery of the matrix block submerged in gas.

The experiment is performed in the following steps:

- The matrix block is saturated with oil. It is usually fully oil-saturated but for other studies it can also be initially saturated with any percentage with oil/water/gas to study different conditions. The initial oil volume should be also measured before saturating the block to know the original oil in place for the matrix block.
- The matrix block is submerged in a pressure container full of water with the desired average pressure as shown in Figure C.2.
- The recovered oil volume is measured periodically.
- The plot of the recovery factor (= the recovered oil volume / the initial oil volume) versus time to create the recovery curve for each case.

C.4 Studying other conditions

That the same procedure as described in Section C.2 and Section C.3 can be repeated for any conditions, for instance under different pressure, fluids (if the fluids type varies in the reservoir) or even applying a pressure gradient to the surrounding fluid to simulate viscous displacement (if it is expected that this happens in the reservoir). Each combination of the driving forces, generates a different curve which should be used for the matrix block ID during the simulation run, if it undergoes the same condition. In other words, it is expected that the same matrix block ID occurs in different parts of the reservoir and in one place it can be in the waterflooded zone while a matrix block of the same ID in some other part of the reservoir is in the gas-invaded zone; therefore, each of them will be depleted on different processes and from different recovery curves that can be pre-measured in the laboratory for the same condition.

C.4.1 Effect of Orientation

The orientation of the block can affect the gravitational driving force if the matrix height and lengths differ. If the orientation of the matrix block ID affects the recovery trend too much, then it can be regarded as two or more *different* matrix block IDs with different recovery curves for each and also later in the simulation, which will be treated as different matrix block IDs. Figure C.3 shows one matrix block shape in two (horizontal and vertical) orientations. Figure C.4 shows the recovery curves for each of these orientations. Since it is the isotropic matrix block just with different orientations, the shape factor and pore volume is the same for both cases and only the gravitational force is changed (by difference of heights) which is the only factor responsible for the recovery curve difference. Therefore, in categorizing the matrix block IDs, these different orientations should be considered as two different matrix block IDs.

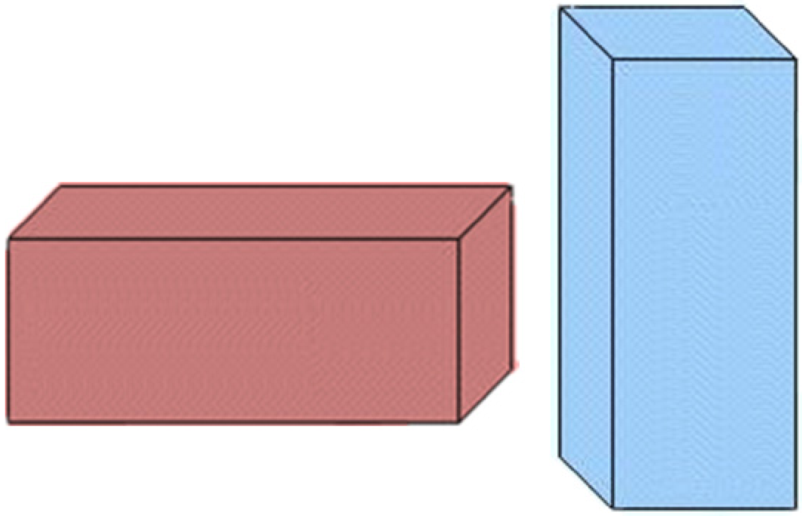


Figure C.3: The same matrix block in different orientations

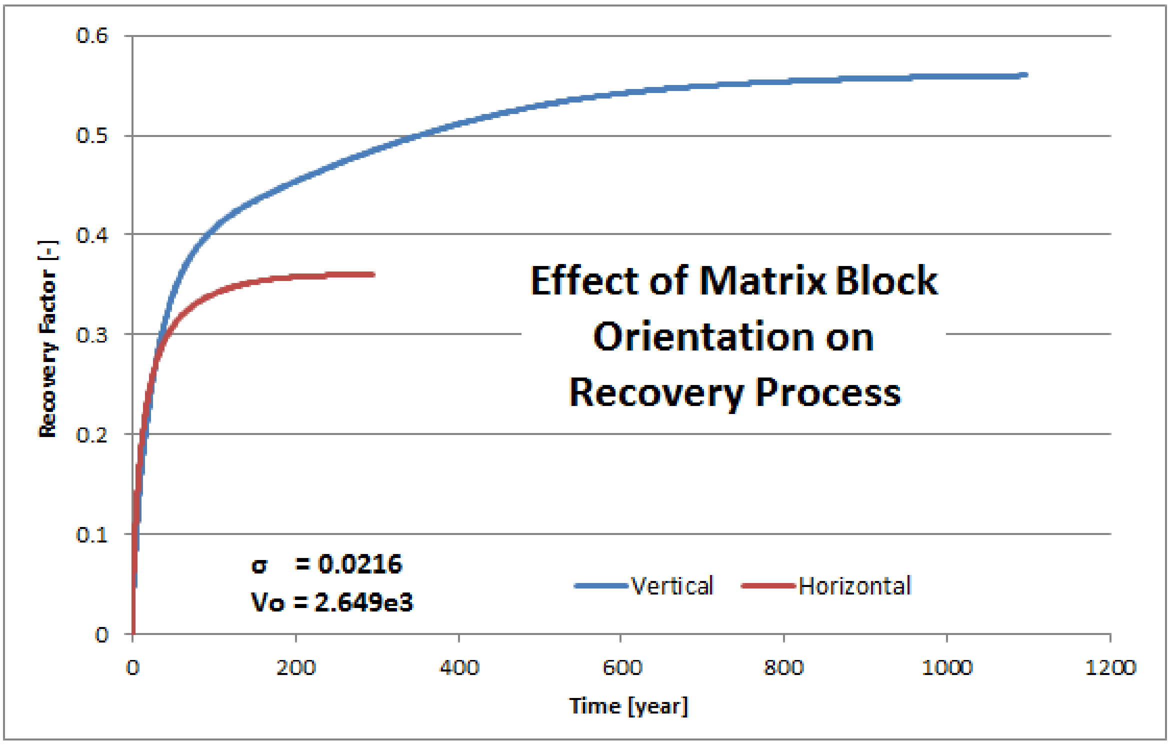


Figure C.4: Recovery Curves of the matrix block IDs in Figure C.3

Appendix D

Implementation of Single Matrix Block Analysis

D.1 Introduction

Single Matrix Block (SMB) analysis is an attempt to numerically simulate the laboratory experiment of studying the matrix block depletion under different conditions and recovery mechanisms.

The laboratory experiment is described in Appendix C and the theory about the proposed SMB model is explained in Section 3.4. In this chapter, the details about the implementation of the proposed Single Matrix Block Analysis model in a commercial reservoir simulator, ***PRS***[45], is explained.

D.2 Objectives

The objective of this section is to introduce a practical and efficient method for performing Single Matrix block calculations. This method should satisfy the following requirements:

- The calculation should be mathematically and numerically correct; e.g. using fully implicit control volume finite difference method on a fine orthogonal grid. The result will be the oil recovery as function of time or dimensionless time.
- The recovery mechanism of a single matrix block should be calculated without any simplifications, by considering all driving forces, pressure and saturation within the matrix and changing states in the surrounding fractures.
- In this work, only black oil models are considered.

D.3 *PRS*[45] Implementation

D.3.1 Problem Explanation

The idea of using Single Matrix Block (SMB) models comes from the ambition to simulate laboratory experiments of studying the matrix block depletion under different conditions and the recovery process for every single *matrix block ID* (Section 4.3.1) and to generate the recovery curve under the set conditions. Such an approach (and laboratory experiment) gives a detailed insight about the processes that are going on between the matrix and the surrounding fractures at different places and under different conditions. This would be the most general and the best approach, if it could be practiced throughout the reservoirs. But unfortunately, this approach is not applicable on the reservoir scale both because it is impossible to access different matrix block IDs (in full) from all around the reservoir to take them to the lab, and that such experiments needs a lot of time and cost to perform and produce the desired outcome. Therefore, SMB laboratory experiments remain restricted only to some research studies and not a standard practice in reservoir modeling.

In general, the experiment (explained in Appendix C in details) is to submerge the matrix block either in water or gas, and collecting the extracted oil and plotting the recovered oil volume versus time. This plot is called a "*Recovery Curve*" which demonstrates how the matrix block under study, depletes in time with the ceratin boundary conditions and active driving mechanisms of the experiment. Knowing the matrix block ID distribution within a given simulation cell, it is possible to accurately estimate the overall recovery curve for the simulation cell (as explained in Section 4.3). As stated earlier, since the actual laboratory experiments are not practical (enough) to be used routinely for the reservoirs, single matrix block models have been developed to numerically simulate such laboratory experiments which makes the study much easier and more affordable.

Different applications of small-scale single matrix block models can be found in the literature[20],[33],[41],[54],[89]. These publications deal with recovery mechanism in fractured reservoirs, primarily with water/oil countercurrent imbibition and gas/oil drainage, and the matrix-fracture transfer. Most of these calculations have been made using commercial simulators placing production and injection wells in the fracture cells. Such a workaround (to model the laboratory experiment) is impractical and in most cases numerically instable, and will produce questionable results with long CPU times. Some of the problems of using commercial simulators to model single matrix blocks are mentioned:

- For every matrix block a new grid model has to be created which requires recreating the grid cells, setting permeability, porosity distributions etc. This is undoubtedly a tedious work and one of the main reasons why such an informative analysis is not even numerically practiced in full field studies.

- The very small pore volumes of the grid cells, especially for the fracture cells causes numerical instability.

- The necessity to use bottom-whole-pressure-controlled production (and/or injection) wells to maintain the fracture conditions causes further numerical instability.

- The boundary conditions for the matrix are determined by the fracture and these cannot be properly set in such a conventional grid model.

- Due to the matrix-fracture fluid transfer, the fluid contents of the fracture changes in time. Assuring constant pressure and saturations in the fracture cells (as in the laboratory experiment of Appendix C) is required for a proper SMB investigation. Setting both pressure and saturations of the fracture cells as constant is in no way achievable in commercial reservoir simulators.

- The influence of viscous displacement cannot be considered in such models because no fluid movement can be incorporated along the fracture cells.

Pirker[81] made some experiments with some commercial software tools, mostly with ECLIPSE[95] and showed that a more sophisticated approach than to model each single matrix block separately is necessary. Pirker[81] tried to standardize this method and developed it to some extent leaving it open with some suggestions. In this work, the single matrix block analysis suggested by Pirker[81] is completed and integrated in a commercial reservoir simulator, ***PRS***[45], which is available to be used for any reservoir model for different matrix block IDs and conditions with minimal effort and can also be implemented in any conventional reservoir simulator.

D.3.2 Solution and Workflow

In order to overcome the problems discussed in Section D.3.1, a new option is proposed that has the following specific advantages and can be implemented very easily in any commercial simulating software. The ***PRS***[45] implementation is briefly described in these steps:

- The grid is automatically set up (based on the actual model) internally and does not require the user to set up a separate grid model for every single matrix block analysis. The great advantage is that no manual preprocessing is required and all the data can be extracted from the full field model and the grid is also automatically generated.

- The time step control and Newton-Ruphson iteration regulation methods were modified so that they work best for micro-scale models (which is neither useful nor included in conventional reservoir simulators) that helps achieve a much better stability which gives the great advantage of being able to calculate even with time steps as large as 180 days; It is worthwhile to mention that using the methods offered in commercial reservoir simulators, the time step lengths can be as short as a fraction of a day. On the other hand, the single matrix blocks usually have to be simulated for several decades or even centuries to be fully depleted! Therefore, being able to calculate with very large time steps accurately enough is really considerable and important.

- Both of the problems of bottom-whole-pressure-controlled wells and variant boundary conditions are overcome by setting back the dynamic fracture properties to their initial values at the end of each time step, after automatically removing all of the fracture oil content from the fractures and reporting it as the amount of extracted oil from the single matrix block.

- The generating recovery curve files that can be readily plotted or used for investigations, study as well as the recovery curve runs are automatically generated.

The following section describes how the single matrix block calculation could be implemented and used in a mature (i.e. in a commercial) reservoir simulator. The coupling of the single matrix calculation to the full field model is essential to enable the interpretation and usage of the single matrix block analysis results directly on the field scale. In this respect other approaches may be possible too, but this description strictly follows the application in ***PRS***[45] (See ***PRS*** Technical Description[47] and ***PRS*** Time Dependent Data Manual[46]).

D.3.3 Activating Single Matrix Block Analysis

The possibility of performing a single matrix block analysis on a real reservoir simulation model, without the necessity to create a new model from scratch with different grid properties for every existing shape factor class, as well as copying different initialization, PVT and rock properties schemes to the new model etc., is really helpful and handy and can shift the single matrix block analysis for a full field from being impossible (or absolutely impractical) to not only possible, but fully automated; this opens a valuable path towards understanding the details about the recovery processes in the reservoir.

PRS[45] provides the single matrix block analysis for any simulation model (currently available only for black-oil) with no effort, just by adding a simple SMBLOCK instruction in the input. If this instruction is used, then the normal run will not be performed and instead, the single matrix block grid is created internally (as described in Section 3.4.2 and Section 3.4.3).

The PVT and ROCK data of the normal model are taken over and the SCHEDULE section and any other unnecessary input (that are related to the actual full field simulation model) will be skipped.

In Section D.3.4, some examples of usages of this instruction are presented.

D.3.4 Practical Grid Construction Alternatives

In addition to the general procedures to construct the grid, described in Section 3.4.2, other practical alternatives can be also imagined for cases that not all the required input exists.

1. Using the matrix block sizes L_x, L_y, L_z in *x-, y-* and *z*-directions respectively. The shape factor will be calculated from Heinemann and Mittermeir[48] formula which is simplified for a cube. If the shape factor is also input in addition to matrix block side lengths, the input shape factor will be discarded and the calculated value is used.
2. Using the shape factor, σ, and the normalized anisotropic permeability values (k'_x, k'_y, k'_z) and the matrix block height, L_z, for which case, it is assumed that L_x, L_y are equal and calculated from Equation D.1 and the shape of the matrix block would be a cuboid.

$$L_x = L_y = \sqrt{\frac{k'_x + k'_y}{\frac{\sigma}{4} - \frac{k'_z}{L_z^2}}}. \tag{D.1}$$

3. Using the shape factor, σ, for which the investigation should be performed and the normalized anisotropic permeability values (k'_x, k'_y, k'_z). In this case L_x, L_y, L_z are assumed to be equal and are calculated from the shape factor as in Equation D.2 and the matrix block would be in shape of a cube.

$$L_x = L_y = L_z = \sqrt{4(k'_x + k'_y + k'_z)/\sigma}. \tag{D.2}$$

D.3.5 Rock and PVT input

In order to assign the rock and PVT input, it is just required to give the PVT region number, PVTREG, the matrix rock region number, ROCKREG, and the fracture rock region number, ROCKREGF keywords in the SMBLOCK instruction.

D.3.6 Porosity and Permeability values

The porosity of the matrix, is input using the keyword PORO, and the porosity of the fracture domain porosity (and therefore its volume) is internally set to *10* times this value. The anisotropic permeability values k_x, k_y, k_z, are input using keywords PERMX, PERMY and PERMZ respectively. There are other keywords to assist entering these values too: PERMXY and PEMRMALL. The fracture cells do not need to be assigned any permeability to (since the transmissibility between two fracture cells, is by default internally set to zero.

D.3.7 Initialization

The equilibrium data of the first equilibrium region of the actual full field simulation model are considered for the single matrix block analysis. The single matrix block model will be initialized based on the input middle-layer PRESSURE and the gravitational gradient in the matrix and fracture separately based on their own fluid saturation.

The matrix is initialized with maximum oil saturation. The fracture is saturated either fully with water, gas or oil for water drive, gas drive or expansion drive mechanism respectively. For oil resaturation study, the fracture is oil-filled and the matrix can have other saturation distribution. In addition to these cases, other combinations of saturation distributions also is possible.

D.3.8 Multi-Case Analysis

In case that different mechanisms and/or shape factors or rock/fluid properties are needed to be tested (which is the usual case in the full-field study process), more than one SMBLOCK instructions can be input in the *.tdd file and ***PRS***[45] performs them one after another automatically. For instance, adding the following instructions in the *.tdd file would sequentially perform 6 SMBLOCK runs and create 6 different output files. Note that the last two are "`lumped`" runs and each will perform more than one SMBLOCK analysis which will be output to the same file. The keywords and attributes are described in details in ***PRS*** TDD Manual[46].

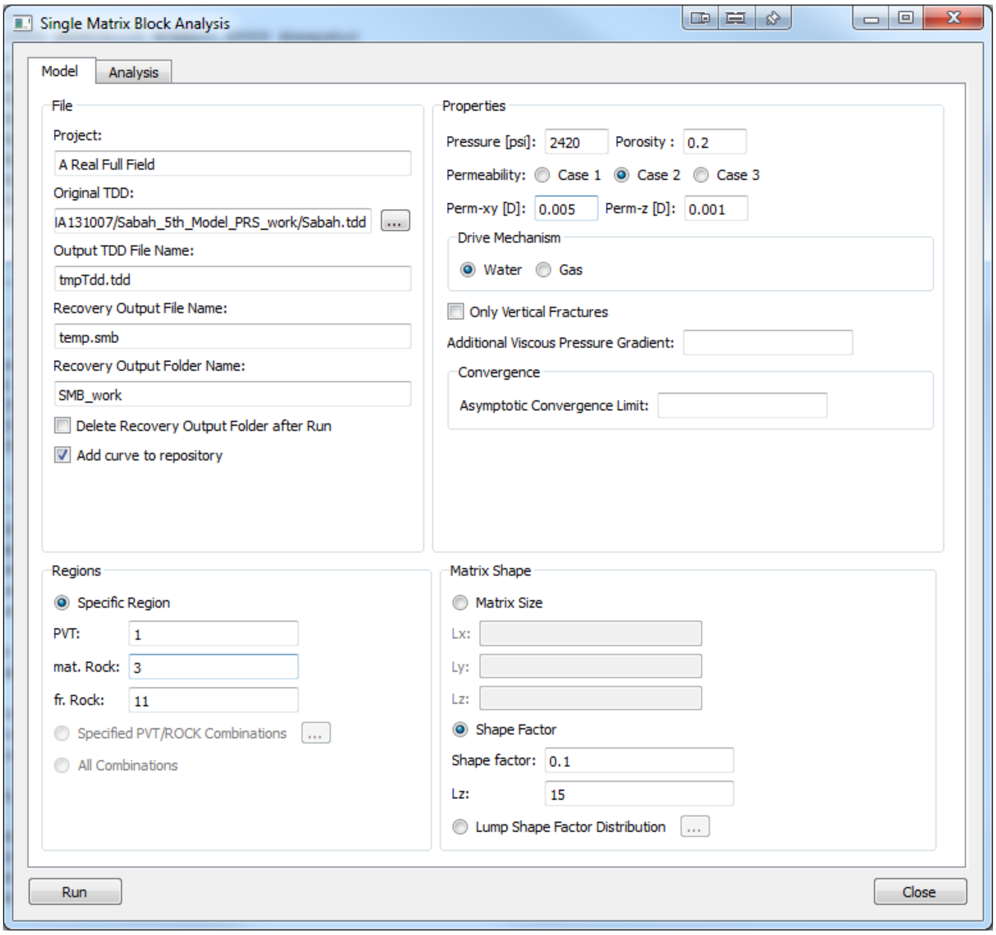
```
# 1
 SMBLOCK  gasdrive   pvtreg 1 rockreg 1 poro 0.2 permall 0.001             &
          rockregF 3     pressure 175 asymplim 1.E-2 lx 20 ly 25 lz 30
# 2
 SMBLOCK  waterdrive pvtreg 2 rockreg 1 poro 0.2 permxy 0.001 permz 0.002 &
          pressgrad 0.05 pressure 175.0 shapefac 0.51 lz 30
# 3
 SMBLOCK  waterdrive pvtreg 1 rockreg 1 poro 0.2 permxy 0.001 permz 0.002 &
          verticalfr     pressure 167.5 shapefac 0.51
# 4
 SMBLOCK  gasrdrive  pvtreg 1 rockreg 1 poro 0.2 permx 0.001 permy 0.012  &
          permz 0.002    pressure 167.5 rockregF 3 shapefac 0.239
# 5
 SMBLOCK  waterdrive pvtreg 1 rockreg 2 poro 0.2 permall 0.001 rockregF 3 &
          verticalfr     pressure 167.5          LZ 30                     &
          lumped 3 shapefac 0.5  0.01 0.05      frequency 0.5  0.2  0.3   &
          FILENAME smblck_recovery1.out
# 6
 SMBLOCK  waterdrive pvtreg 1 rockreg 1 poro 0.2 permall 0.001            &
          pressure 167.5 lumped 2 shapefac 0.01 0.01  LZ  30  25
```

D.4 Graphical User Interface

After development of the Single Matrix Block concept and implementing it in ***PRS***, the need of a user-friendly tool which offers the full range of advantages of the recovery curve approach to simulation of NFRs in practice was felt. Thus, in the course of this study a comprehensive user interface was developed as an auxiliary option in ***PRS*** which can be readily used in full-field studies and is here briefly introduced. The ***SMBAux*** tool works on full field model setup in ***PRS*** to 1) perform Single Matrix Block Analyses under desired conditions, 2) offer a repository for the calculated recovery curves, 3) plotting recovery curves and scaled recovery curves, 4) calculating class recovery curves, 5) lumping recovery curves and 6) creating region recovery curves.

D.4.1 Single Matrix Block Analysis

A single matrix block analysis can be easily performed for a full-field model using ***SMBAux*** tool. On the `Model` tab, Figure D.1, there are four sections to input the necessary data to perform the SMB run and a `Run` button in the bottom which actually starts ***PRS***.



Single Matrix Block Analysis

Model | Analysis

File

Project: A Real Full Field

Original TDD: IA131007/Sabah_5th_Model_PRS_work/Sabah.tdd ...

Output TDD File Name: tmpTdd.tdd

Recovery Output File Name: temp.smb

Recovery Output Folder Name: SMB_work

□ Delete Recovery Output Folder after Run

☑ Add curve to repository

Properties

Pressure [psi]: 2420 Porosity : 0.2

Permeability: ○ Case 1 ● Case 2 ○ Case 3

Perm-xy [D]: 0.005 Perm-z [D]: 0.001

Drive Mechanism

● Water ○ Gas

□ Only Vertical Fractures

Additional Viscous Pressure Gradient:

Convergence

Asymptotic Convergence Limit:

Regions

● Specific Region

PVT: 1

mat. Rock: 3

fr. Rock: 11

○ Specified PVT/ROCK Combinations ...

○ All Combinations

Matrix Shape

○ Matrix Size

Lx:

Ly:

Lz:

● Shape Factor

Shape factor: 0.1

Lz: 15

○ Lump Shape Factor Distribution ...

Run

Close

Figure D.1: Single Matrix Block Analysis Tool - Model Tab

D.4.1.1 File Section

In the file section, the main input file of the full field model to the ***PRS*** can be browsed and the ***SMBAux*** tool would used this to access the PVT and SCAL data of the full field.

The names of the temporary files and folders which would be created for SMB calculation can also be modified in this section. Checking `Add curve to repository`, means that the calculated recovery curve should be added to the ***SMBAux*** repository for plotting and further operations. Otherwise, ***SMBAux*** runs ***PRS*** with the appropriate input settings to calculate the requested SMB analysis and the output would be found in the folder specified by `Recovery Output Folder Name` input field and the file with the name specified in `Recovery Output File Name` field.

D.4.1.2 Regions Section

In this section, the PVT region serial number and the Rock regions serial numbers of the matrix and fractures are input based on the full-field input.

D.4.1.3 Properties Section

In this section, the mandatory input of initial average `pressure` for the matrix block is input. This pressure is assigned to the mid-layer in both matrix and fracture and the pressure of the other layers would be calculated based on the gravitational pressure gradient based on the initial fluid saturations respectively in the matrix and the fracture cells. Another optional pressure gradient can be applied in the fracture domain in addition to the gravitational pressure gradient to model the viscous forces using `Additional Viscous Pressure Gradient` field.

The porosity and directional `permeability` values of the matrix block are also input in this section.The `drive mechanism` can be chosen as either water drive or gas drive which means that the fracture would be filled with water or gas respectively. If `Only Vertical Fractures` is checked, the horizontal fracture cells would become inactive.

D.4.1.4 Matrix Shape Section

In this section, the matrix block size is defined using one of these methods:

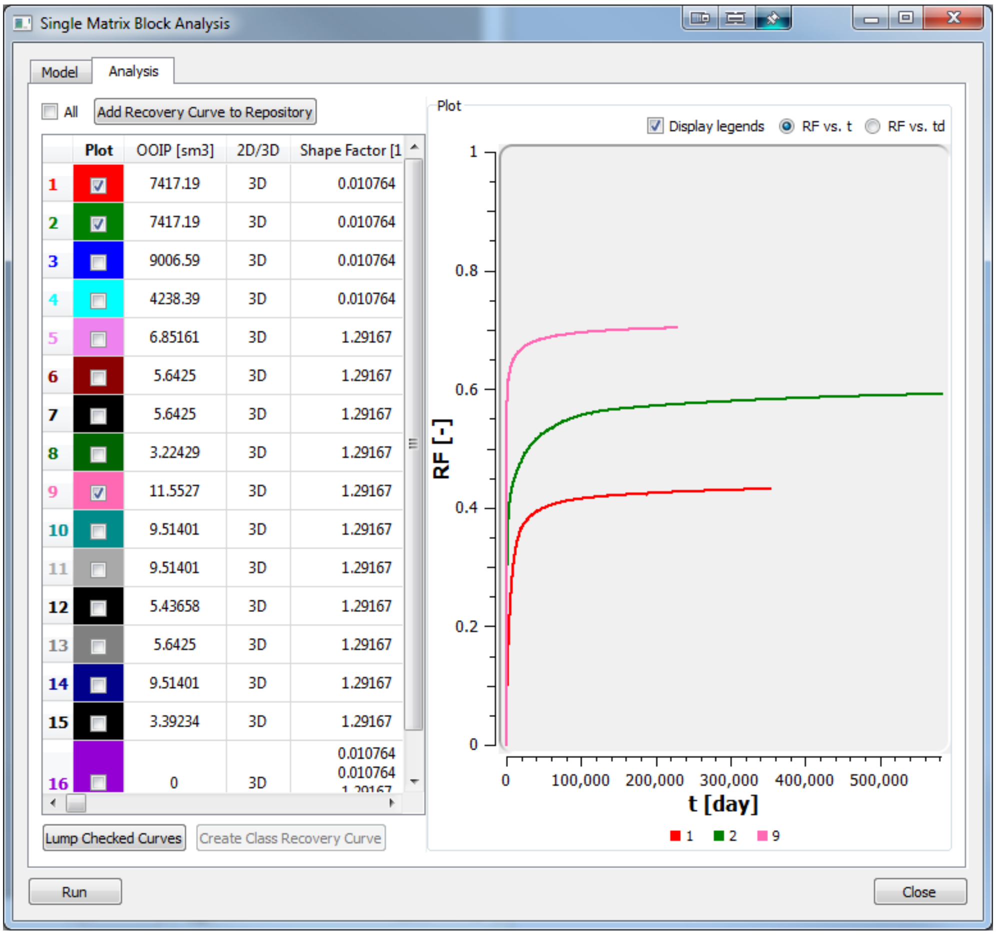
- Inputting the side lengths of the matrix block cuboid `Lx`, `Ly`, `Lz`.
- Inputting the `shape factor` and optionally the matrix block height, `Lz`, and the matrix block side lengths will be calculated internally as previously explained.

In this section, another option also exists which will be explained in Section D.4.3.

D.4.2 Recovery Curve Repository

On the `Analysis` tab, Figure D.2, there are two main sections on each side of the screen. On the left side, a list of all the SMB analyses in the repository exists. A checkbox exists for each SMB analysis and checking it, activates the according SMB analysis and plots it in the right-side plot section.

Using the `Add Recovery Curve to Repository` button, an SMB Analysis output from ***PRS*** which was not calculated using ***SMBAux*** tool can also be added to the repository.



☑ Display legends ● RF vs. t ○ RF vs. td

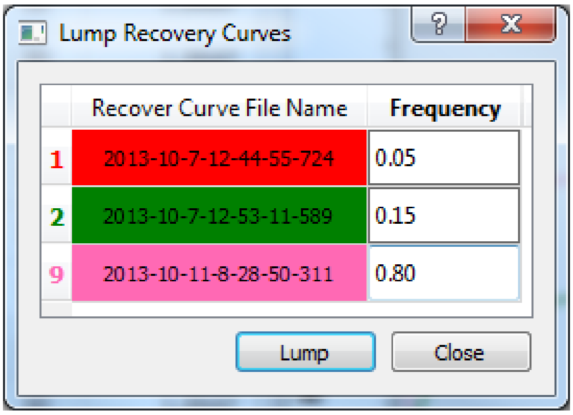
	Plot	OOIP [sm3]	2D/3D	Shape Factor [1
1	☑	7417.19	3D	0.010764
2	☑	7417.19	3D	0.010764
3	□	9006.59	3D	0.010764
4	□	4238.39	3D	0.010764
5	□	6.85161	3D	1.29167
6	□	5.6425	3D	1.29167
7	□	5.6425	3D	1.29167
8	□	3.22429	3D	1.29167
9	☑	11.5527	3D	1.29167
10	□	9.51401	3D	1.29167
11	□	9.51401	3D	1.29167
12	□	5.43658	3D	1.29167
13	□	5.6425	3D	1.29167
14	□	9.51401	3D	1.29167
15	□	3.39234	3D	1.29167
16	□	0	3D	0.010764 0.010764 1.29167

Figure D.2: Single Matrix Block Analysis Tool - Analysis Tab

On the top of the plot section, it can be selected if the recovery factor should be plotted versus time (`RF vs. t`) or dimensionless time (`RF vs. td`) which is the scaled recovery curve; and this is applied to all of the curves on the plot.

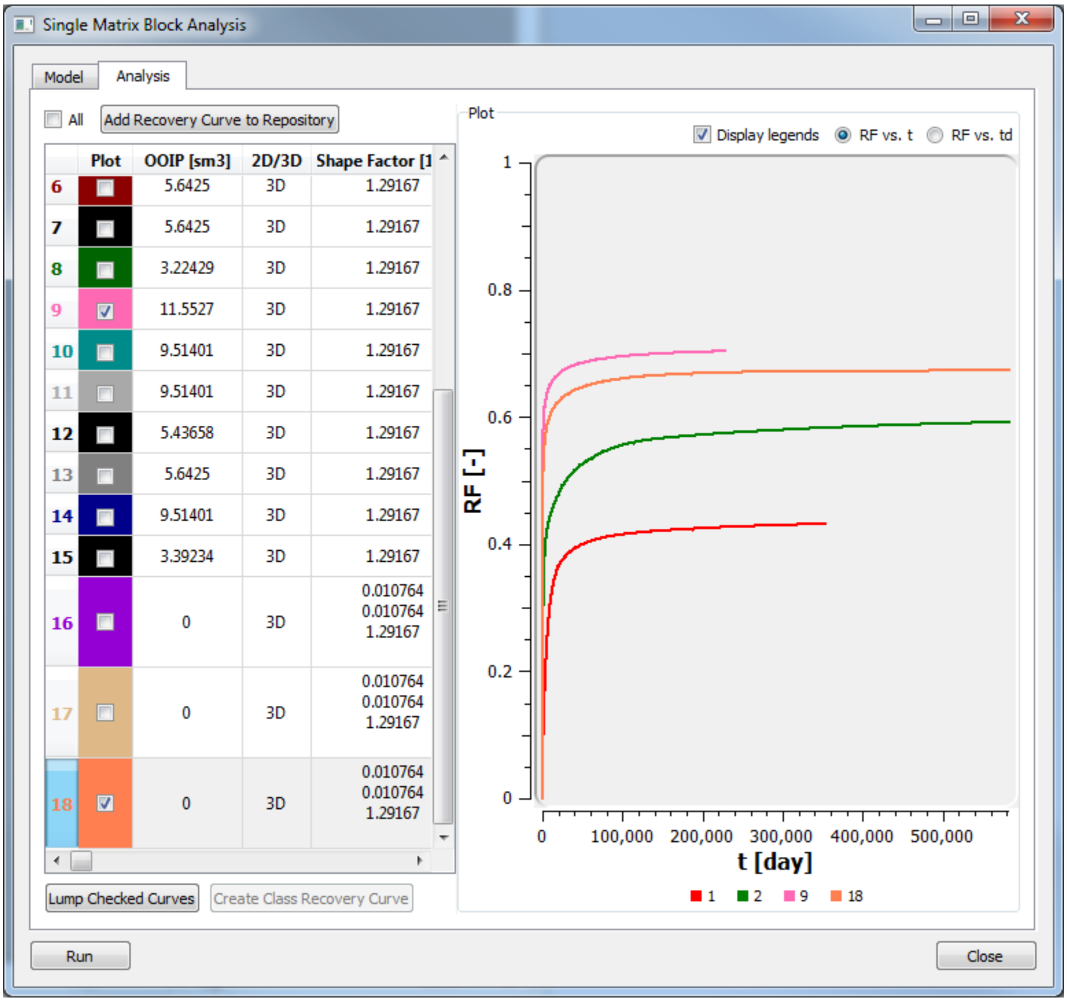
D.4.3 Lumping Recovery Curves

It is possible to lump the recovery curves (Figure 3.9) by checking the desired recovery curves and clicking the `Lump Checked Curves` button which pops up the window of Figure D.3.



	Recover Curve File Name	Frequency
1	2013-10-7-12-44-55-724	0.05
2	2013-10-7-12-53-11-589	0.15
9	2013-10-11-8-28-50-311	0.80

Figure D.3: Lump Recovery Curves window



	Plot	OOIP [sm3]	2D/3D	Shape Factor [1
6	□	5.6425	3D	1.29167
7	□	5.6425	3D	1.29167
8	□	3.22429	3D	1.29167
9	☑	11.5527	3D	1.29167
10	□	9.51401	3D	1.29167
11	□	9.51401	3D	1.29167
12	□	5.43658	3D	1.29167
13	□	5.6425	3D	1.29167
14	□	9.51401	3D	1.29167
15	□	3.39234	3D	1.29167
16	□	0	3D	0.010764 0.010764 1.29167
17	□	0	3D	0.010764 0.010764 1.29167
18	☑	0	3D	0.010764 0.010764 1.29167

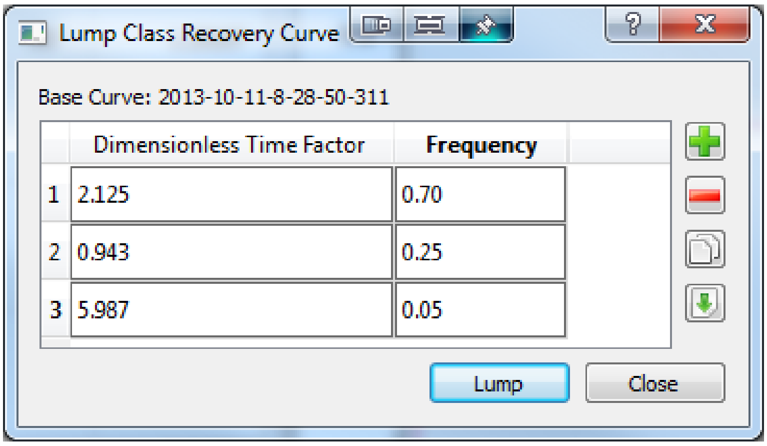
Figure D.4: Plot of the Lumped Curve (orange) and the curves it was calculated from

In this window, a list of all the selected curves exists (in Figure D.3, the three selected curves plotted in Figure D.2 can be seen). The frequency of each curve should be input in the field to the right of its title; the summation of the entered frequencies has to be 1. After clicking the `Lump` button, ***SMBAux*** lumps the curve (by calculating the new recovery factor values from Equation 3.23 at any time) and adds the lumped curve to the repository where it can be plotted as shown in Figure D.4.

Note that the lumped curve continues in time as much as the longest of the curves being lumped (e.g. the green curve in Figure D.2) and the values of the curves which end earlier for lumping after their finish time is taken as their ultimate recovery since this is their actual recovery also after the finishing time. In Figure D.4 the orange curve is the lumped curve of the three selected curves in Figure D.2 with the frequencies displayed in Figure D.3.

D.4.4 Preparing Class Recovery Curve

If the SMB analysis of one matrix block belonging to a certain class is known, it is possible to calculate the class recovery curve explained in Section 5.2.1.3.1 by checking the desired SMB Analysis recovery curve (only one curve can be checked at this time) and the `Create Class Recovery Curve` button enables. The checked recovery curve from the repository would be called the *Base Curve* from now on. Clicking this button opens Figure D.5.



Base Curve: 2013-10-11-8-28-50-311

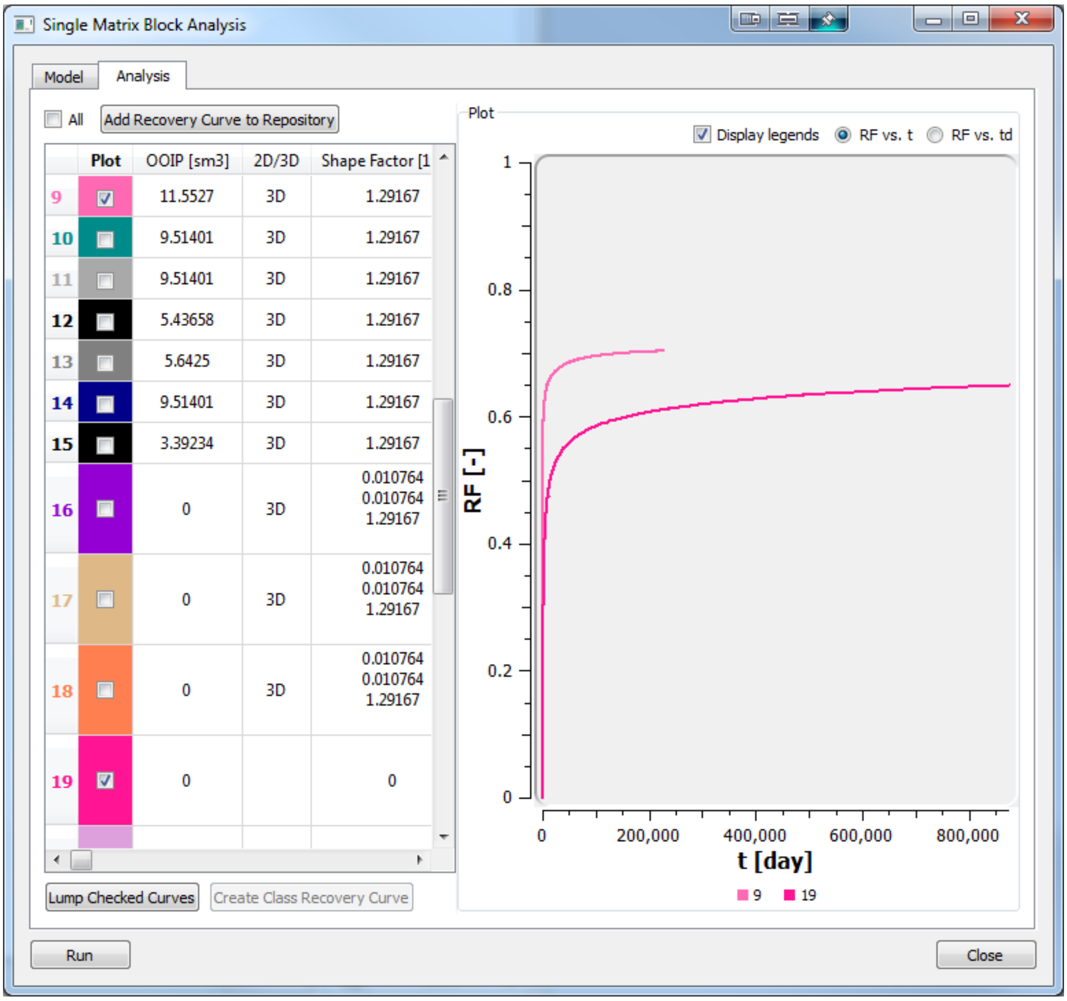
	Dimensionless Time Factor	Frequency
1	2.125	0.70
2	0.943	0.25
3	5.987	0.05

Figure D.5: Lump Class Recovery Curve window

In Figure D.5 any desired number of rows can be added using the + button and the values can be input directly or imported from comma-separated-values file or pasted from clipboard. The table has two columns: the first column is the dimensionless time conversion factor (β from Equation 3.26) of all the matrix block IDs (see Section 4.3.1) in the desired recovery curve class; the second column is the frequency of each matrix block ID in that class. After clicking the `Lump` button, the following operations are performed one after the other one:

- For each input matrix block ID (i.e. each row in Figure D.5), a new recovery curve is internally created where its time axis values are equal to the dimensionless time values from the base curve divided by the input dimensionless time factor for that matrix block ID and the same recovery factor values as the base curve.

- These curves are lumped with the frequencies entered for each with the same procedure as in Section D.4.3 resulting in a new recovery curve.
- The calculated recovery curve is actually the class recovery curve of a class with the scaled recovery curve as the base curve and the entered distribution of matrix block IDs. The class recovery curve will be added to repository and is shown in Figure D.6.



	Plot	OOIP [sm3]	2D/3D	Shape Factor [1
9	☑	11.5527	3D	1.29167
10	☐	9.51401	3D	1.29167
11	☐	9.51401	3D	1.29167
12	☐	5.43658	3D	1.29167
13	☐	5.6425	3D	1.29167
14	☐	9.51401	3D	1.29167
15	☐	3.39234	3D	1.29167
16	☐	0	3D	0.010764 0.010764 1.29167
17	☐	0	3D	0.010764 0.010764 1.29167
18	☐	0	3D	0.010764 0.010764 1.29167
19	☑	0		0

Figure D.6: Plot of the lumped Class Recovery Curve (flushed pink #19) and the plot of the Base Curve (pale pink #9)

D.4.5 Creating Region Recovery Curves

In order to create the region recovery curves, nothing more is required than to check the class recovery curves created in Section D.4.4 and lump them based on their distribution in every region exactly in the same manner described in Section D.4.3.

The last column in the left side grid in the `Analysis` tab is `File Name in Repository` which is self explanatory and all the recovery curves in the repository can be accessed in the **SMB_Repository** folder in separate files with *.smb extension.

Appendix E

Production Mechanisms in Naturally Fractured Reservoirs

E.1 Introduction

In this chapter, the mechanisms that take part in extraction of hydrocarbons from the matrix to the fracture are described.

The same processes that are active in nonfractured, single porosity systems are important in fractured reservoirs. These include rock compaction, fluid expansion, viscous drive, gravity displacement, capillary imbibition and diffusion. However, the majority of the oil is contained in the high porosity but low permeability matrix system while the wells drain the fluids directly from the high permeability fractures. Therefore, the degree of importance of the physical mechanisms may be quite different in the two systems.

Production from the matrix blocks can be associated with various physical mechanisms including:

- Rock compressibility and compaction,
- Single phase fluid expansion,
- Solution gas drive,
- Gravity drainage,
- Capillary Imbibition,
- Diffusion,
- Viscous displacement.

E.2 Expansion and Solution Gas Drive

As the pressure drops in the fracture system, as a consequence of production, fluid expands and flows out from the matrix to equilibrate the matrix pressure with the surrounding fracture pressure. Also the compressibility of rock or the secondary compaction can gain importance especially low fluid compressibility or low porosity. The pore compressibilities for matrix and fracture are in most of the cases quite different.

Below the bubble point pressure the solution gas liberates and expands. The efficiency of the matrix recovery under solution gas displacement can be calculated in a standalone way, using the well-known Muskat[65] or Pirson[82] methods. According to Pirson[82], the equations that describe the performance of the reservoir in case of solution gas drive reservoir can be written in finite difference form of the material balance:

$$\Delta\left[\frac{N_p}{N}\right] = \frac{\left[1-\frac{N_p}{N}\right]\Delta\left[\frac{B_o}{B_g}-R_s\right]-[1+m_{g,o}]B_{ob}\Delta\frac{1}{B_g}}{\left[\frac{B_o}{B_g}-R_s\right]_{i+1}+\bar{R}} \quad . \tag{E.1}$$

The instantaneous gas-oil ratio is:

$$R = R_s + \frac{B_o}{B_g}\cdot\frac{k_{rg}}{k_{ro}}\cdot\frac{\mu_o}{\mu_g} \quad . \tag{E.2}$$

The oil saturation is

$$S_o = \left[1-\left(\frac{N_p}{N}\right)\right]\cdot\frac{B_o}{B_{ob}}(1-S_w) \quad . \tag{E.3}$$

Muskat[65] used also a differential form of the material balance equation to evaluate the performance of a reservoir with internal gas-drive (solution gas drive). The change in the oil saturation is:

$$\Delta S_o = \Delta p\left[\frac{S_o\lambda(p)+S_o\frac{k_g}{k_o}\eta(p)+(1-S_w-S_o)\varepsilon(p)}{\omega(p,S_o)}\right] , \tag{E.4}$$

where $\lambda(p)$, $\varepsilon(p)$, $\eta(p)$ and $\omega(p,S_o)$ are special Muskat[65] functions:

$$\lambda(p) = \frac{B_g}{B_o}\frac{dR_s}{dp} , \tag{E.5}$$

$$\varepsilon(p) = B_g \frac{d}{dp}\left(\frac{1}{B_g}\right) , \tag{E.6}$$

$$\eta(p) = \frac{1}{B_o} \frac{\mu_o}{\mu_g} \frac{dB_o}{dp} , \tag{E.7}$$

$$\omega(p, S_o) = 1 + \frac{k_g \mu_o}{k_o \mu_g} . \tag{E.8}$$

The cumulative amount of produced oil is then:

$$N_p = V_p \sum_{p_i}^{p} \Delta\left(\frac{S_o}{B_o}\right) = V_p\left[\left(\frac{S_o}{B_o}\right)_i - \left(\frac{S_o}{B_o}\right)\right] . \tag{E.9}$$

The method proposed by Muskat[65] is advantageous if a larger number of calculations is required. The Muskat[65] functions (Equation E.5 - Equation E.8) are readily available in form of curves (see Figure E.1).

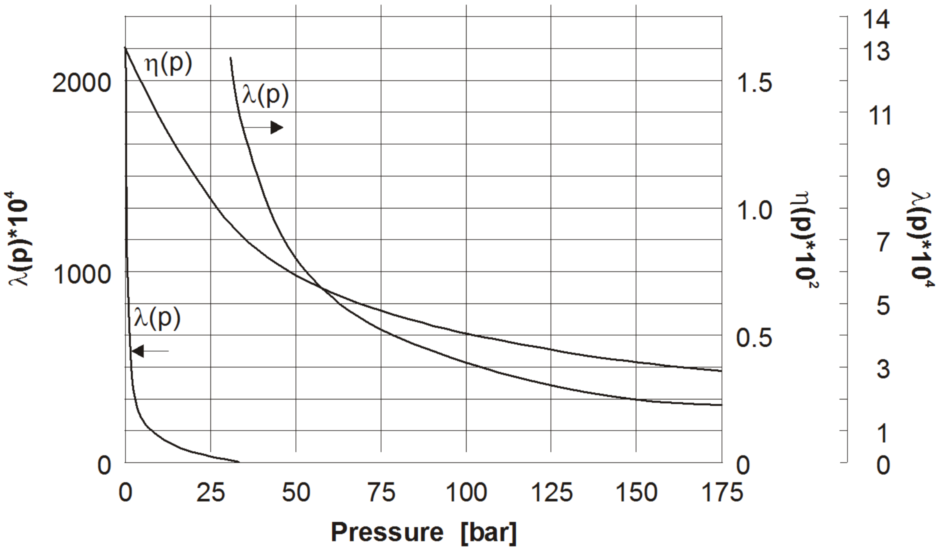


Figure E.1: Muskat functions[65]

Muskat and Taylor[66] analyzed the influence of the oil viscosity, the initial pressure, the solution GOR, the connate water saturation and the relative permeability ratio on oil production.

The pressure and GOR will be developed similar to those in Figure E.2, calculated by Muskat *et al.*[65]. The method can be applied to a single matrix block and to the whole matrix-fracture system as well. Note that in the first case the reservoir pressure and the recovery factor is valid for the matrix, and not for the entire porosity system including the fractures. The numerical calculations of expansion mechanism are very accurate on every scale (also in full field models), while it is free of time truncation, numerical dispersion and grid orientation errors.

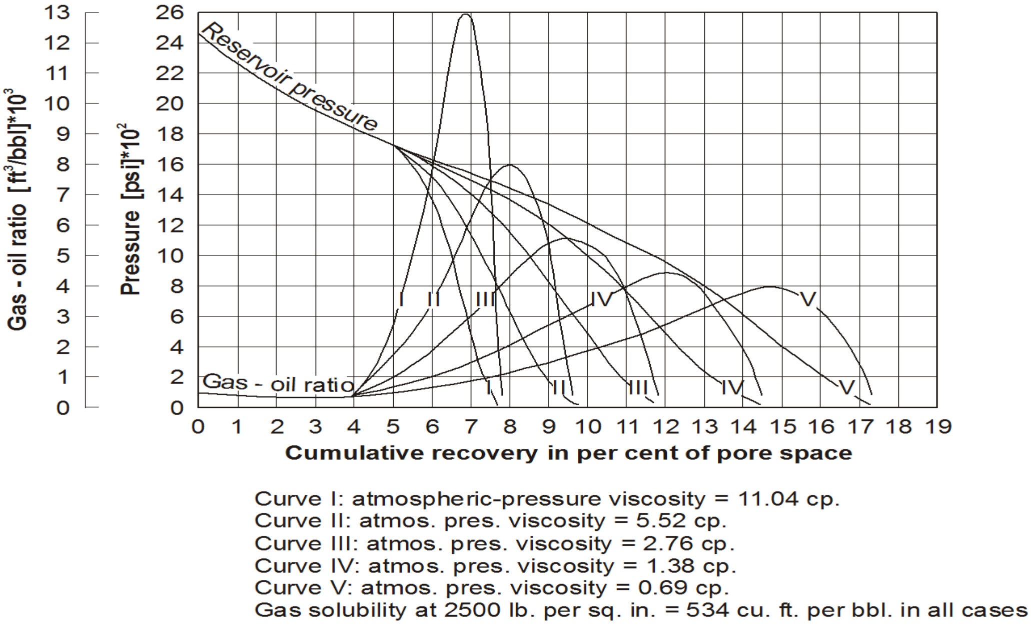


Figure E.2: Pressure and gas oil ratio histories of solution gas-drive reservoirs producing oil of different viscosities (after Muskat and Taylor[66])

E.3 Capillary Imbibition

Let's imagine a piece of porous rock, called matrix, surrounded by fractures. The matrix is assumed to be water wet. The water invading the fractures flows, driven by capillary forces, into the matrix, displacing oil in countercurrent way, which means that water imbibes the matrix from the same side where oil is expelled. Depending on the fracture configuration, the displacement has two or three dimensional character, but it can be examined on a simplified one dimensional model.

Now let's consider a core sealed on all sides except the bottom surface, as shown in Figure E.3. The core is saturated by oil and the initial water saturation is equal with the connate water saturation S_{wm}. At the bottom side the core is contacted with the wetting phase. Due to capillary forces the wetting phase, the water tends to intrude at the bottom side and thus displaces the nonwetting phase, the oil, in counterflow.

To calculate the displacement process the following assumptions will be made:

1. The rock is homogenous.
2. The displacement is one dimensional.
3. The two phases are incompressible and immiscible.
4. Multi-phase Darcy's equation is valid.

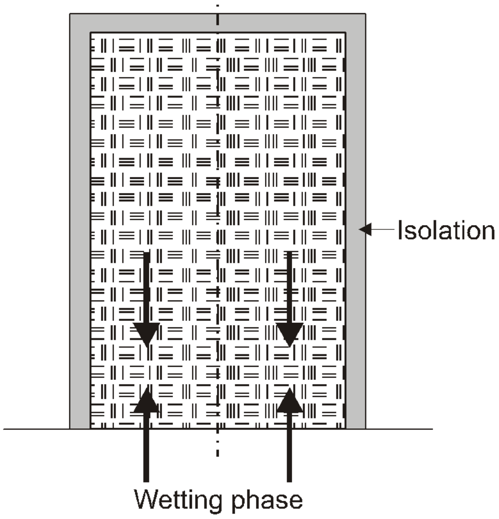


Figure E.3: Countercurrent imbibition

As consequence of these assumptions the overall fluid velocity must be everywhere zero:

$$u_w + u_o = 0 \quad , \tag{E.10}$$

where

$$u_w = -\frac{kk_{rw}}{\mu_w}\left(\frac{\partial p_w}{\partial x} + \rho_w g\right) \quad , \tag{E.11}$$

and

$$u_o = -\frac{kk_{ro}}{\mu_o}\left(\frac{\partial p_o}{\partial x} + \rho_o g\right) \quad . \tag{E.12}$$

The difference of the phase pressures is the capillary pressure:

$$p_o - p_w = P_c(S_w) \quad . \tag{E.13}$$

The continuity equation for the water and oil are:

$$\frac{\partial u_w}{\partial x} + \phi\frac{\partial S_w}{\partial t} = 0 \quad , \tag{E.14}$$

$$\frac{\partial u_o}{\partial x} + \phi\frac{\partial S_o}{\partial t} = 0 \quad . \tag{E.15}$$

The saturation constraint is:

$$S_w + S_o = 1 \quad . \tag{E.16}$$

Then Equation E.11 is divided by k_{rw}/μ_w and Equation E.12 by k_{ro}/μ_o. Afterwards, they are subtracted one from another:

$$u_w\left(\frac{\mu_w}{k_{rw}} + \frac{\mu_o}{k_{ro}}\right) = k(\rho_w - \rho_o)g + k\frac{dP_c}{dS_w} \cdot \frac{\partial S_w}{\partial x} \quad , \tag{E.17}$$

or written it in a more compact form:

$$u_w = \varphi_w + \psi_w \cdot \frac{\partial S_w}{\partial x} \quad , \tag{E.18}$$

where

$$\varphi_w = \frac{k(\rho_w - \rho_o)g}{\dfrac{\mu_w}{k_{rw}} + \dfrac{\mu_o}{k_{ro}}} \quad , \tag{E.19}$$

$$\psi_w = \frac{k}{\dfrac{\mu_w}{k_{rw}} + \dfrac{\mu_o}{k_{ro}}} \cdot \frac{dP_c}{dS_w} \quad . \tag{E.20}$$

Substituting of Equation E.18 into Equation E.14 leads to

$$\frac{\partial}{\partial x}\left[\varphi_w + \psi_w \frac{\partial S_w}{\partial x}\right] + \phi \frac{\partial S_w}{\partial t} = 0 \quad , \tag{E.21}$$

or

$$\frac{1}{\phi}\left[\frac{\partial \varphi_w}{dS_w} \cdot \frac{\partial S_w}{\partial x} + \frac{\partial}{\partial x}\left(\psi_w \frac{\partial S_w}{\partial x}\right)\right] + \frac{\partial S_w}{\partial t} = 0 \quad . \tag{E.22}$$

The boundary conditions are:

At the outlet *x = L:*

$$u_w = \left[\varphi_w + \psi_w \frac{\partial S_w}{\partial x}\right]_{x = L} = 0 \quad . \tag{E.23}$$

At the inlet $x = 0$, the capillary pressure is zero, thus:

$$(S_w)_{x=0} = S_{wM} = 1 - S_{or}, \tag{E.24}$$

where S_{or} is the residual oil saturation and S_{wM} is the maximum possible water saturation. The solution of the above boundary value problem (Equation E.22 – Equation E.24) is the saturation as a function of the location and time:

$$S_w = S_w(x, t), \tag{E.25}$$

which can be found just by applying numerical methods.

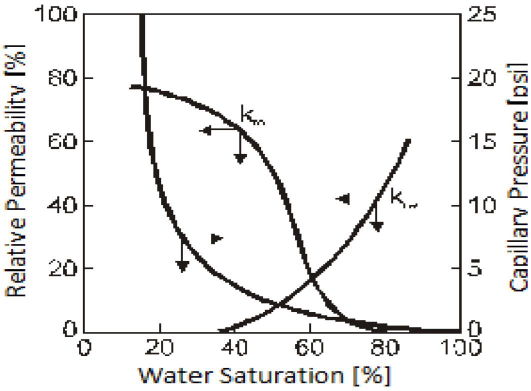


Figure E.4: Capillary pressure and relative permeability functions used in the calculations (Blair[11]).

Blair[11] presented numerical solution for the model defined by Figure E.3. Figure E.4 shows the capillary pressure and relative permeability curves used in the calculations. The results showed similarity to a concurrent linear displacement, which means that a relative narrow transition zone, interpreted as water front, was formed between the displacing and displaced phases.

Figure E.5 shows the pressure and saturation profile when the water front has reached approximately half the length of the sample. The water saturation is high at the open end of the core, but decreases then abruptly. The pressure gradient in the water phase is highest near the front. The greatest pressure gradient in the oil phase occurs near the open end of the core, because of the fact that with the maximum oil flow rate the oil relative permeability is the minimum. In case of strongly water wet rock the process of spontaneous imbibition lasts theoretically until the oil saturation in the matrix reaches its residual value. The results of the calculations showed that the rate of imbibition depends strongly on rock and fluid properties.

The numerical results from Blair[11] were verified by experiments of Graham and Richardson[38]. They measured the oil production as function of the time. One result is shown in Figure E.6 where all parameters, except the core lengths, were constant. The results give a feeling to the time dependence of the imbibition process.

The results presented above are strictly valid for one dimensional displacement of two incompressible fluids. The matrix recovery processes under real conditions are more complicated, and therefore the applicability of the Blair[11] model is limited. General mathematical models, as used in reservoir simulation practise, make it possible to investigate the mechanisms in fully complexity.

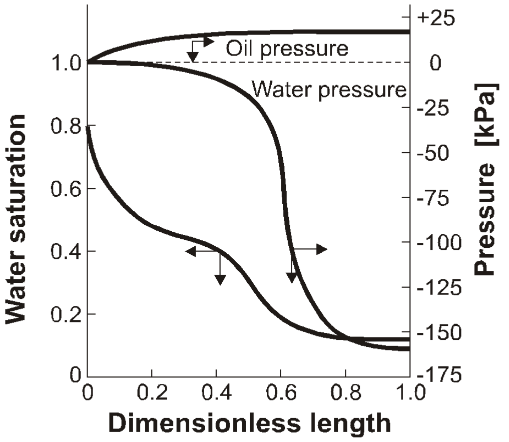


Figure E.5: Distribution of pressure and saturation after 6.6 hours. (by Blair[11]).

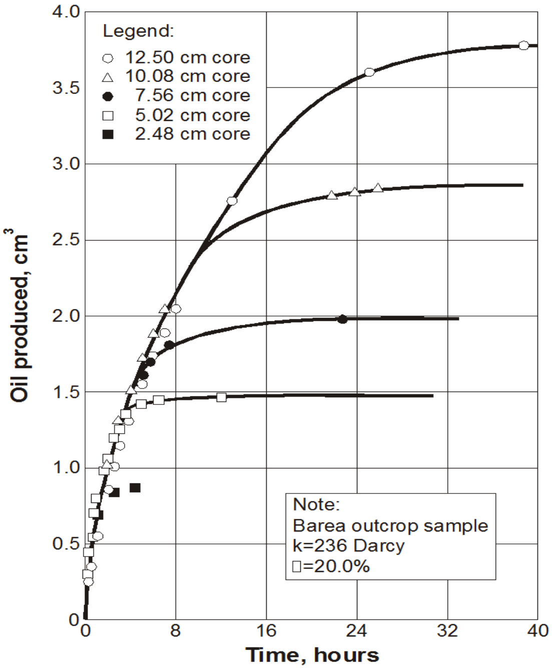


Figure E.6: Recovery in case of linear counterflowing imbibition and the experimental determination of the influence of a certain in core-length. (by Graham and Richardson[38])

Mattax and Kyte[62] demonstrated that the recovery behavior of a large reservoir matrix block can be predicted from an imbibition test on a small core sample. The oil recovery can be scaled for a given rock type and oil-to-water ratio by a dimensionless rate parameter:

$$t = \sqrt{\frac{k}{\phi}}\frac{\sigma_{ow}}{\mu_w L^2} \quad , \tag{E.26}$$

where t is the imbibition time, k is the permeability, ϕ is the porosity, σ_{ow} is the interfacial tension, μ_w is the water viscosity and L is a characteristic length for a matrix block. From Equation E.26 can be concluded that in an imbibition displacement the time of recovery is proportional to the square of the block height, L. Mattax and Kyte[62] showed that if the imbibition oil recovery is plotted against the dimensionless scaling parameter the same recovery curve will be obtained for the model and all reservoir matrix blocks of the same rock type and geometry:

$$\left(t\sqrt{\frac{k}{\phi}}\frac{\sigma}{\mu_w L^2}\right)_{\text{model}} = \left(t\sqrt{\frac{k}{\phi}}\frac{\sigma}{\mu_w L^2}\right)_{\text{matrix block}} \quad . \tag{E.27}$$

Figure E.7 shows the results of the verification of the scaling parameter.

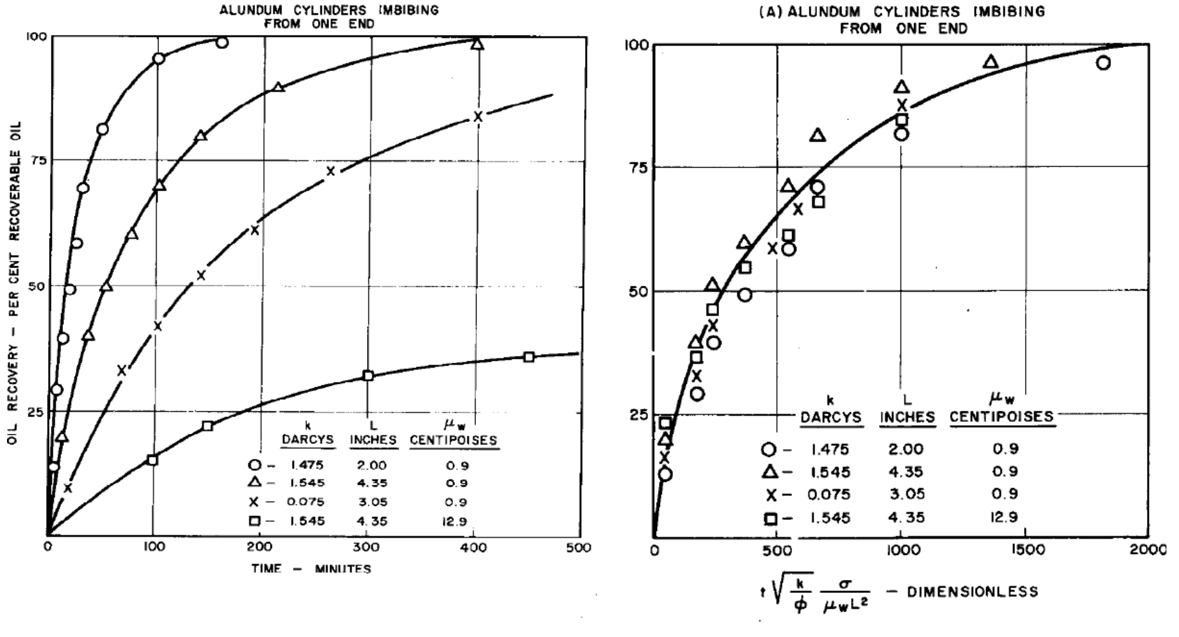


Figure E.7: Verification of imbibition scaling[62]

In gas-oil systems the oil is the wetting phase and tends to imbibe into the matrix. In practice this means in absence of gravity drainage no oil production occurs from the matrix block when the associated fracture block is filled by gas.

E.4 Gravity Drainage

When the fluid contents of the matrix and fracture are not the same (e.g. fracture is filled with gas or water), a difference in the hydrostatic head due to gravitational force exists between the matrix and the fracture. This additional potential difference, may force water or gas from the fracture into the matrix and expel the oil out of the matrix; this process is called gravity drainage.

E.4.1 Gravity drainage assuming homogeneous vertical saturation

Regard a matrix block surrounded by vertical fractures. The matrix is saturated by oil and the fracture filled by (a) water or (b) by gas. In absence of capillary forces there would no difference between the phase pressures: $p = p_o = p_g = p_w$. The pressures at the middle depth of the matrix block and in the fracture are in a pseudo-steady state situation equal. During a transient process such as well testing, the pressures could be different at this point but then they will be equalized within a short time. The different hydrostatical gradients create a pressure difference above and bellow the midpoint as it is shown in Figure E.8.

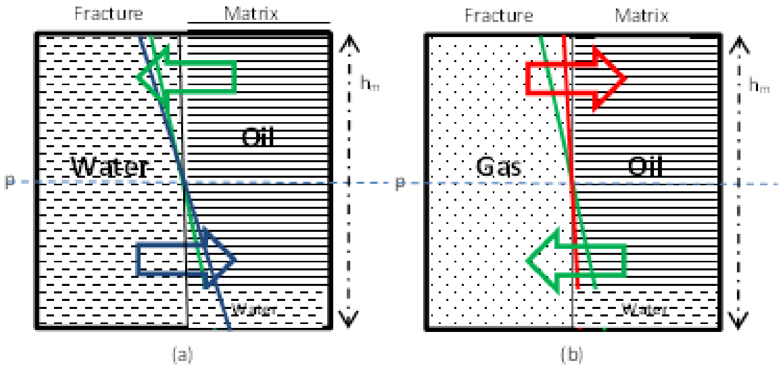


Figure E.8: Vertical pressure distributions in matrix and in fracture (a) filled with water, (b) filled with gas

In the case that the fracture is filled by water the water will be enter in the matrix at the lower half and the oil expels at the upper one. The average hydrostatic pressure difference between the matrix and the fracture are the same for both phases:

$$P_{hwof} = \frac{1}{2}h_m(\rho_w - \rho_o)g \quad , \tag{E.28}$$

where h_m is the vertical extension of the matrix block. This pressure difference act on the half length of the matrix block which has to be considered while calculating the effective potential differences for the water and for the oil phases:

$$\Phi_{wm} - \Phi_{wf} = p_{om} - p_{of} - P_{cwom} + P_{cwof} - \frac{1}{4}P_{hwof} , \tag{E.29}$$

$$\Phi_{om} - \Phi_{of} = p_{om} - p_{of} + \frac{1}{4}P_{hwof} . \tag{E.30}$$

Similar for the gas filled fracture case:

$$P_{hgof} = \frac{1}{2}h_m(\rho_g - \rho_o)g , \tag{E.31}$$

and the potential differences for the gas and for the oil phases:

$$\Phi_{gm} - \Phi_{gf} = p_{om} - p_{of} + P_{cogf} - P_{cogf} - \frac{1}{4}P_{gof} , \tag{E.32}$$

$$\Phi_{om} - \Phi_{of} = p_{om} - p_{of} + \frac{1}{4}P_{hogf} . \tag{E.33}$$

The gas enter at the upper half of the matrix and the oil expels at the lower one. Above it was assumed that the fracture is entirely filled by water or gas. In this case the fracture capillary pressures are zero per definition:

$$P_{cwof} = P_{cogf} = 0 . \tag{E.34}$$

This remains valid also if no *100%* water or gas but a homogeneous vertical fluid saturation in the fracture is still assumed.

E.4.2 Gravity drainage assuming phase segregation

E.4.2.1 Two-phase water-oil case

Figure E.9 illustrates an idealized matrix block surrounded by fractures. Both the matrix and corresponding fractures contain oil and water. Assuming perfect phase separation both in the fracture and in the matrix, the hydrostatic pressure difference can be calculated for the matrix and for the fracture:

$$P_{hwof} = h_{wf}(\rho_w - \rho_o)g , \tag{E.35}$$

$$P_{hwom} = h_{wm}(\rho_w - \rho_o)g . \tag{E.36}$$

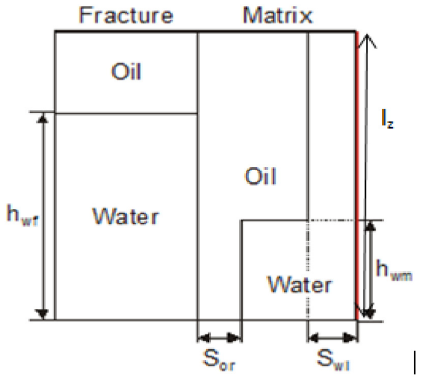


Figure E.9: Matrix block partially merged in water

The difference in the hydrostatic head is an additional driving force for the interporosity flow and must be considered:

$$\Phi_{wm} - \Phi_{wf} = p_{om} - p_{of} - P_{cwom} + P_{cwof} + P_{hwom} - P_{hwof} \quad , \tag{E.37}$$

$$\Phi_{om} - \Phi_{of} = p_{om} - p_{of} - P_{hwom} + P_{hwof} \quad . \tag{E.38}$$

The hydrostatic heads $h_f$and h_m can be expressed using the saturations. For the fracture:

$$h_{wf} = L_z \cdot \left(\frac{S_w - S_{wi}}{1 - S_{wi} - S_{orw}} \right)_f \approx L_z \cdot S_{wf} \quad , \tag{E.39}$$

and for the matrix:

$$h_{wm} = L_z \cdot \left(\frac{S_w - S_{wi}}{1 - S_{wi} - S_{orw}} \right)_m \quad . \tag{E.40}$$

Inserting Equation E.39 and Equation E.40 into Equation E.36 and Equation E.35 the hydrostatic pressure differences become functions of the saturation too, and so they are formally similar to the well-understood capillary pressure:

$$P_{hwof}(S_{wf}) = L_z \cdot S_{wf} \cdot (\rho_w - \rho_o) g \quad , \tag{E.41}$$

$$P_{hwom}(S_{wm}) = L_z \cdot \left(\frac{S_w - S_{wi}}{1 - S_{wi} - S_{orw}} \right)_m (\rho_w - \rho_o) g \quad . \tag{E.42}$$

For this reason, the hydrostatic pressure difference usually is referred to as pseudo-capillary pressure and the fluid exchange induced by the gravitation as gravity imbibition and drainage.

E.4.2.2 Two phase gas-oil case

Figure E.10 illustrates an idealized matrix block surrounded by fractures. Both the matrix and corresponding fractures contain oil and gas. Similarly to the water/oil case, the following is valid for oil and gas:

$$P_{hgof}(S_{gf}) = L_z \cdot S_{gf} \cdot (\rho_o - \rho_g)g \quad , \tag{E.43}$$

$$P_{hgom}(S_{gm}) = L_z \cdot \left(\frac{S_g}{1 - S_{wi} - S_{org}}\right)_m (\rho_o - \rho_g)g \quad ; \tag{E.44}$$

$$\Phi_{gm} - \Phi_{gf} = p_{om} - p_{of} + P_{cogm} - P_{cogf} + P_{hgom} - P_{hgof} \quad , \tag{E.45}$$

$$\Phi_{om} - \Phi_{of} = p_{om} - p_{of} - P_{hgom} + P_{hgof} \quad . \tag{E.46}$$

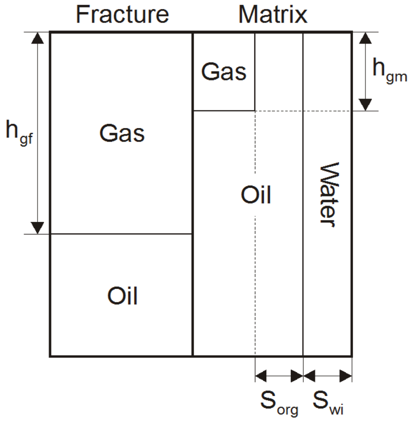


Figure E.10: Matrix block partially invaded by gas.

E.4.2.3 Quandalle and Sabathier gravity drainage model

An alternative gravity drainage model was suggested by Quandalle and Sabathier[84]. The matrix-fracture flow is taken to be the sum of three flows from the center of the matrix to the fractures: one horizontal one vertical upwards and one vertically downwards[95] as shown in Figure E.11. Theoretically this model has the following advantages:

- The shape factor can be different in vertical and horizontal directions,
- The vertical anisotropy can be considered,
- The gravity force (in vertical direction) can be considered,
- The relative permeability depends on the direction.

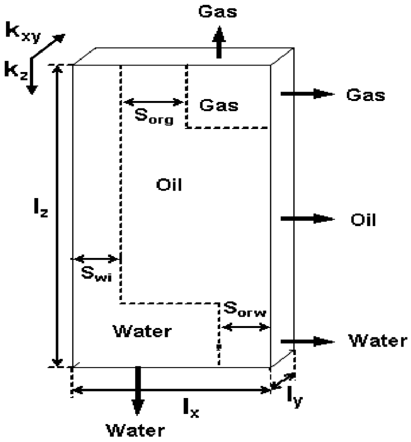


Figure E.11: Modified gravity drainage model (after Quandalle and Sabatier[84])

Nevertheless, the practical advantage of this method is questionable. There is no way to determine the necessary parameters for the Quandalle and Sabatier[84] approach in a reliable way. On top of that, to calculate the lateral oil, water and gas rates (showed by horizontal arrows in Figure E.11) requires an estimate for the apparent fluid density in the fracture. While this can be done explicitly[84], the numerical stability of method is reduced and therefore the applicable time step length is limited.

E.5 Viscous Displacement

Consider a representative matrix cell, shown in Figure E.12. The average pressure in the matrix cell and in the fracture will be equalized due of the capillary and gravitational forces. In the isolated matrix cell the fluid does not move therefore, the pressure is uniform. Contrarily, there is a pressure gradient in the fracture system while the fluid is moving towards the production wells (and away from the injection wells). Therefore, there is a pressure difference Δp between the two edges of the cell.

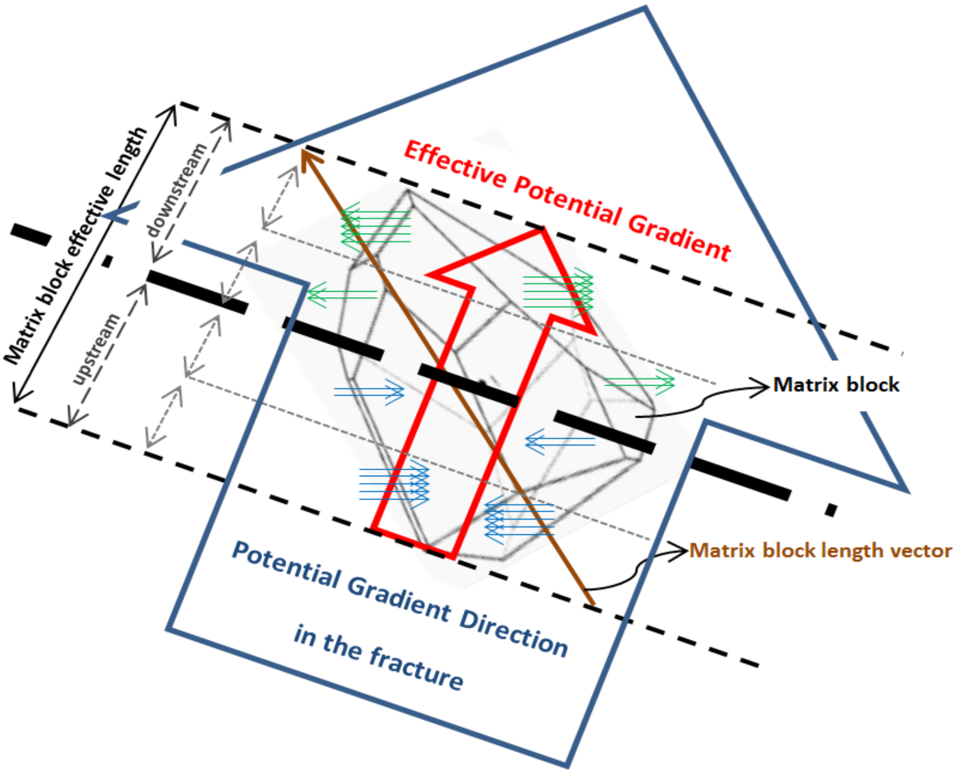

Effective Potential Gradient
Matrix block effective length
downstream
upstream
Matrix block
Matrix block length vector
Potential Gradient Direction
in the fracture

Figure E.12: Schematic of the viscous effect in fluid exchange between the matrix block (in the center) and the surrounding fracture

A pressure difference of $\Delta p/4$ acts in average at the upstream half and $-\Delta p/4$ at the downstream half of the matrix block. Consequently, fluid enters from the fracture into the matrix at the upstream half and flows out from the matrix to the fracture at the downstream half. This additional driving force can contribute to the matrix recovery process.

The pressure gradient for the phase *p* at the grid point *i* is

$$\vec{\nabla\Phi}_p = \Phi'_{px}\vec{i} + \Phi'_{py}\vec{j} + \Phi'_{pz}\vec{k} \quad , \tag{E.47}$$

where Φ' are the partial derivatives. The potential difference between the grid point *i* and the neighbor *j* is given by the scalar product of the neighbor's position vector

$$\vec{r}_{ij} = \Delta x_{ij}\vec{i} + \Delta y_{ij}\vec{j} + \Delta z_{ij}\vec{k} \quad , \tag{E.48}$$

and the potential gradient. In the case of *N* neighboring grid points a linear equation system of *N* equations with 3 unknowns can be created:

$$\Phi'_{px}\Delta x_{ij} + \Phi'_{py}\Delta y_{ij} + \Phi'_{pz}\Delta z_{ij} = \Phi_{pj} - \Phi_{pi} \quad , j = 1, N. \tag{E.49}$$

To achieve the three components of the potential gradient $\Phi'_{p\alpha}$; $\alpha = x, y, z$ the Gauss normal equations has to created and solved:

$$\begin{aligned}
\Phi'_{px}\sum\Delta x_{ij}^2 + \Phi'_{py}\sum\Delta y_{ij}\Delta x_{ij} + \Phi'_{pz}\sum\Delta z_{ij}\Delta x_{ij} &= \sum(\Phi_{pj} - \Phi_{pi})\Delta x_{ij} \\
\Phi'_{px}\sum\Delta x_{ij}\Delta y_{ij} + \Phi'_{py}\sum\Delta y_{ij}^2 + \Phi'_{pz}\sum\Delta z_{ij}\Delta y_{ij} &= \sum(\Phi_{pj} - \Phi_{pi})\Delta y_{ij} \\
\Phi'_{px}\sum\Delta x_{ij}\Delta z_{ij} + \Phi'_{py}\sum\Delta y_{ij}\Delta z_{ij} + \Phi'_{pz}\sum\Delta z_{ij}^2 &= \sum(\Phi_{pj} - \Phi_{pi})\Delta z_{ij}
\end{aligned} \qquad \text{(E.50)}$$

where the summation is over the neighbors $j=1,...,N$. Assuming that the matrix block has the size of L_x, L_y, L_z, the potential difference between matrix and fracture at the two sides of the matrix block is:

$$\Delta\Phi_{pv} = \frac{1}{4}\overrightarrow{\nabla\Phi_p} \bullet (L_x\vec{i} + L_y\vec{j} + L_z\vec{k}) \ . \qquad \text{(E.51)}$$

The potential difference at the grid point is

$$\Delta\Phi_{pmf} = \Phi_{pm} - \Phi_{pf} \ . \qquad \text{(E.52)}$$

If $\Delta\Phi_{pmf} + \Delta\Phi_{pv} \ldots and \ldots \Delta\Phi_{pmf} - \Delta\Phi_{pv}$, have the same sign then at both half of the matrix block either the matrix or the fracture is the upstream side and the term $\Delta\Phi_{pv}$ will be cancelled out in the flow equation, consequently the viscous force has no effect.

If both $\Delta\Phi_{pmf} + \Delta\Phi_{pv} > 0$ and $\Delta\Phi_{pmf} - \Delta\Phi_{pv} < 0$ then the phase p enters the matrix block at the upstream side (if $\lambda_{pf} > 0$) and expels at the downstream one.

Finally, the matrix fracture fluid exchange rate for the phase p with considering of the viscous force is:

$$q_{mfp} = \frac{\tau_{mf}}{2}\lambda_{pf}(\Delta\Phi_{pmf} + \Delta\Phi_{pv}) - \frac{\tau_{mf}}{2}\lambda_{pm}(\Delta\Phi_{pmf} - \Delta\Phi_{pv}) \ . \qquad \text{(E.53)}$$

Note that the Equation E.53 is applicable only if the mentioned inequalities are valid. In the derivation above, no restriction regarding the positions of the neighboring grid point were made, so the approach can be applied on unstructured grid too. The disadvantage is that the calculation time is very high.

A simpler approach can be applied only on a structured grid as it is used by ECLIPSE[95] using a viscous displacement model, based on a technique described by Gilman and Kazemi[36].

E.6 Molecular Diffusion

In naturally fractured reservoirs, as opposed to unfractured reservoirs, molecular diffusion may be very important, because the dispersive flux through fractures rapidly increases the contact area for diffusion. Similarly, Fick's molecular diffusion potential may even override viscous forces when hydrocarbon or inert gases are injected and the fracture spacing is small.

The following method is used to calculate the diffusion between fracture and matrix blocks. However, it is not used for grid blocks within the same grid system. The method can also be used to calculate gas-gas and gas-liquid diffusion rates. This approach was described by da Silva and Belery[24].

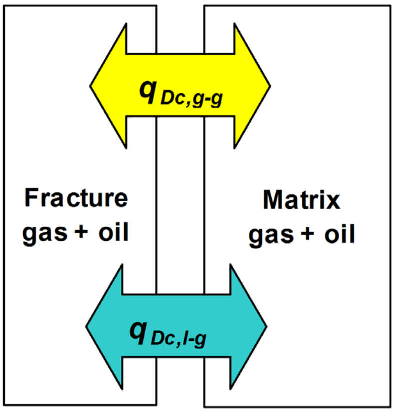


Figure E.13: Schematic of gas-gas and liquid-gas diffusion

When hydrocarbon gas is injected at pressures below the oil bubble point pressure, two-phase diffusion can take place. For example, the injected gas dispersed throughout the fractures may contact large areas with two-phase saturated matrix blocks. Due to concentration differences, the injected gas tends to diffuse through both hydrocarbon phases, which simultaneously cause counterflow to the fractures. However, the gas-gas diffusion is approximately ten times faster than the gas-liquid diffusion.

The mixing process observed during multicomponent fluid displacements in porous media is typically of convection-diffusion type. Convection arises from the heterogeneity of the medium inducing local bulk velocity differences, while diffusion refers to the random motion of the molecules. The combination of the two types of molecular motion causes dispersion (i.e., a mechanism which will tend to abrogate any spatial concentration differences). In principle, the dispersive flux of a particular fluid component depends upon the concentration gradients of all other components. The use of effective coefficients eliminates all the cross terms. The following simulation approach can be proposed based on the work described by Coats[21] and da Silva and Belery[24]:

Diffusion is only calculated between matrix and fracture blocks (and not between blocks that are lying in the same grid system). Only the gas-gas diffusion or the gas-liquid diffusion is taken into consideration (if both are taken into consideration, an equilibrium-state dilemma arises which would require all matrix component K-values to be equal).

Furthermore, the effective diffusion coefficient must be known for each component. The diffusion coefficients are calculated from a static variable group as reservoir temperature, component molecular weights and critical properties of the components and from dynamic variables as pressure and composition. When dealing with black-oil models it is only possible to calculate the liquid-gas diffusion rate since the gas phase consists of a single component. In compositional simulation more phases exist, therefore the gas-gas diffusion rate can also be calculated. If, for example, gas with a different composition than the reservoir gas is injected into a reservoir the gas-gas diffusion should be taken into account. Figure E.13 shows a schematic of gas-gas and liquid-gas diffusion between a matrix and a fracture grid block. The matrix-fracture diffusion transfer is given for gas-gas diffusion as

$$q_{Dc,\,g-g} = \tau_D S_{gm} j_{gc}[(D_g x_{gc})_m - (D_g x_{gc})_f] \cdot (1 - S_{wf}) \ , \tag{E.54}$$

and for the gas-liquid diffusion as

$$q_{Dc,\,o-g} = \tau_D S_{om} j_{oc}[(D_o x_{oc})_m - (D_g x_{gc})_f] \cdot (1 - S_{wf}) \text{, with} \tag{E.55}$$

$$\tau_D = V_m \cdot 8\left[\frac{1}{L_x^2} + \frac{1}{L_y^2} + \frac{1}{L_z^2}\right] \cdot \frac{\phi_m}{\tau_{or}} \ , \tag{E.56}$$

where

$q_{Dc,\,g-g}$ - gas-gas diffusion rate for component *c* [mol/d],
$q_{Dc,\,o-g}$ - liquid-gas diffusion rate for component *c* [mol/d],
τ_D - diffusion transmissibility [m],
j_{gc} - effective gas diffusion coefficient for component *c* [m^2/d],
j_{oc} - effective liquid diffusion coefficient for component *c* [m^2/d],
τ_{or} - tortuosity (porous medium correction factor) [-].

The multiplication by (1-S_{wf}) (Equation E.54 and Equation E.55) is performed to prevent the requirement for a diffusion calculation for immersed matrix blocks by water. The contribution of the diffusion will be added to the matrix-fracture transfer term.